

マシンオートメーションコントローラ NJ シリーズ

スタートアップガイド CPUユニット編

Startup
Guide

おことわり

- (1) 本ガイドの内容の一部または全部を無断で複写、複製、転載することを禁じます。
- (2) 本ガイドの内容に関しては、改良のため予告なしに仕様などを変更することがあります。あらかじめご了承ください。
- (3) 本ガイドの内容に関しては、万全を期しておりますが、万一ご不審な点や誤りなど、お気づきのことがありましたら、お手数ですが、弊社支店または営業所までご連絡ください。その際、巻末記載の**Man.No.** (マニュアルNo.) も併せてお知らせください。

商標

- Sysmacは、オムロン株式会社FA機器製品の日本およびその他の国における商標または登録商標です。
- Windows、Windows XP、Windows Vista、Windows 7、Windows 8は、米国Microsoft Corporationの米国およびその他の国における登録商標です。
- Intel、インテル、Intel ロゴ、インテル Atom は、米国およびその他の国におけるインテル コーポレーションの商標です。
- EtherCAT®は、ドイツBeckhoff Automation GmbHによりライセンスされた特許取得済み技術であり登録商標です。
- ODVA、CIP、CompoNet、DeviceNet、EtherNet/IPはODVAの商標です。
- SD、SDHCロゴは、SD-3C, LLCの商標です。 

その他、本文中に掲載しているシステム名および製品名は、それぞれ各社の商標または、登録商標です。

はじめに

このたびは、NJ シリーズ CPU ユニットおよび Sysmac Studio をお買い上げいただきまして、ありがとうございます。

NJ シリーズ スタートアップガイド CPU ユニット編（以降「本ガイド」と省略します。）は、NJ シリーズ CPU ユニートを初めて使用する場合の立ち上げ手順と Sysmac Studio の基本的な操作方法を、簡単なシーケンス制御の例で説明しています。本ガイドに沿って操作することで、NJ シリーズ CPU ユニットと Sysmac Studio の使い方の基礎を短時間で習得できます。また、NJ シリーズ CPU ユニットと Sysmac Studio を使用するうえで、知っておくと便利な機能を紹介しています。

なお、本ガイドには安全に関する注意事項を含め、使用する際の詳細内容については記載していません。必ず、本ガイドで使用する各機器のマニュアルや取扱説明書を入手し、「安全上のご注意」「安全上の要点」「使用上の注意」などの安全に関する注意事項を含め、使用する際の詳細内容を確認したうえで使用してください。

モーション制御を行う場合の手順と操作については、『NJ シリーズ スタートアップガイド モーション制御編（SBCE-372）』を参照してください。

対象となる読者の方々

本ガイドは、次の方を対象に記述しています。

電気の知識（電気工事士あるいは同等の知識）を有する方で

- FA 機器の導入を担当される方
- FA システムを設計される方
- FA 機器を設置、接続される方

対象となる製品

本ガイドは、以下の製品を対象にしています。

- マシンオートメーションコントローラ NJ シリーズ CPU ユニット
- オートメーションソフトウェア Sysmac Studio

アイコン

本ガイドで使用するアイコンには、次のような意味があります。



安全上の要点

製品を安全に使用するために実施または回避すべきことを示します。



使用上の注意

製品の動作不能、誤動作、または性能・機能への悪影響を予防するために実施または回避すべきことを示します。



参考

必要に応じて読んでいただきたい項目です。

知っておくと便利な情報や、使用する上で参考となる内容について説明しています。



詳しい情報や、関連情報の記載されている場所を示します。

ご承諾事項

マシンオートメーションコントローラ NJ シリーズ CPU ユニット

「当社商品」について特別の合意がない場合には、お客様のご購入先にかかわらず、本ご承諾事項記載の条件を適用いたします。

● 定義

本ご承諾事項中の用語の定義は次のとおりです。

- ・「当社商品」：「当社」の FA システム機器、汎用制御機器、センシング機器、電子・機構部品
- ・「カタログ等」：「当社商品」に関する、ベスト制御機器オムロン、電子・機構部品総合カタログ、その他のカタログ、仕様書、取扱説明書、マニュアル等であって電磁的方法で提供されるものも含まれます。
- ・「利用条件等」：「カタログ等」に記載の、「当社商品」の利用条件、定格、性能、動作環境、取り扱い方法、利用上の注意、禁止事項その他
- ・「お客様用途」：「当社商品」のお客様におけるご利用方法であって、お客様が製造する部品、電子基板、機器、設備またはシステム等への「当社商品」の組み込み、または利用を含みます。
- ・「適合性等」：「お客様用途」での「当社商品」の (a) 適合性、(b) 動作、(c) 第三者の知的財産の非侵害、(d) 法令の遵守および (e) 各種規格の遵守

● 記載事項のご注意

「カタログ等」の記載内容については次の点をご理解ください。

- ・定格値および性能値は、単独試験における各条件のもとで得られた値であり、各定格値および性能値の複合条件のもとで得られる値を保証するものではありません。
- ・参考データはご参考として提供するもので、その範囲で常に正常に動作することを保証するものではありません。
- ・利用事例はご参考ですので、「当社」は「適合性等」について保証いたしかねます。
- ・「当社」は、改善や当社都合等により、「当社商品」の生産を中止し、または「当社商品」の仕様を変更することがあります。

● ご利用にあたってのご注意

ご採用およびご利用に際しては次の点をご理解ください。

- ・定格・性能ほか「利用条件等」を遵守しご利用ください。
- ・お客様ご自身にて「適合性等」をご確認いただき、「当社商品」のご利用の可否をご判断ください。「当社」は「適合性等」を一切保証いたしかねます。
- ・「当社商品」がお客様のシステム全体の中で意図した用途に対して、適切に配電・設置されていることをお客様ご自身で、必ず事前に確認してください。
- ・「当社商品」をご使用の際には、(i) 定格および性能に対し余裕のある「当社商品」のご利用、冗長設計などの安全設計、(ii) 「当社商品」が故障しても、「お客様用途」の危険を最小にする安全設計、(iii) 利用者に危険を知らせるための、安全対策のシステム全体としての構築、(iv) 「当社商品」および「お客様用途」の定期的な保守、の各事項を実施してください。
- ・「当社」は DDoS 攻撃（分散型 DoS 攻撃）、コンピュータウイルスその他の技術的な有害プログラム、不正アクセスにより、「当社商品」、インストールされたソフトウェア、またはすべてのコンピュータ機器、コンピュータプログラム、ネットワーク、データベースが感染したとしても、そのことにより直接または間接的に生じた損失、損害その他の費用について一切責任を負わないものとします。

お客様ご自身にて、①アンチウイルス保護、②データ入出力、③紛失データの復元、④「当社商品」またはインストールされたソフトウェアに対するコンピュータウイルス感染防止、⑤「当社商品」に対する不正アクセス防止についての十分な措置を講じてください。

- ・「当社商品」は、一般工業製品向けの汎用品として設計製造されています。従いまして、次に掲げる用途での使用は意図しておらず、お客様が「当社商品」をこれらの用途に使用される際には、「当社」は「当社商品」に対して一切保証をいたしません。ただし、次に掲げる用途であっても「当社」の意図した商品用途の場合や特別の合意がある場合は除きます。
 - (a) 高い安全性が必要とされる用途（例：原子力制御設備、燃焼設備、航空・宇宙設備、鉄道設備、昇降設備、娯楽設備、医用機器、安全装置、その他生命・身体に危険が及ぶ用途）
 - (b) 高い信頼性が必要な用途（例：ガス・水道・電気等の供給システム、24 時間連続運転システム、決済システムほか権利・財産を取扱う用途など）
 - (c) 厳しい条件または環境での用途（例：屋外に設置する設備、化学的汚染を被る設備、電磁的妨害を被る設備、振動・衝撃を受ける設備など）
 - (d) 「カタログ等」に記載のない条件や環境での用途
- ・上記の (a) から (d) に記載されている他、「本カタログ等記載の商品」は自動車（二輪車含む。以下同じ）向けではありません。自動車に搭載する用途には利用しないで下さい。自動車搭載用商品については当社営業担当者にご相談ください。

● 保証条件

「当社商品」の保証条件は次のとおりです。

- ・保証期間 ご購入後 1 年間といたします。
（ただし「カタログ等」に別途記載がある場合を除きます。）
- ・保証内容 故障した「当社商品」について、以下のいずれかを「当社」の任意の判断で実施します。
 - (a) 当社保守サービス拠点における故障した「当社商品」の無償修理
（ただし、電子・機構部品については、修理対応は行いません。）
 - (b) 故障した「当社商品」と同数の代替品の無償提供
- ・保証対象外 故障の原因が次のいずれかに該当する場合は、保証いたしません。
 - (a) 「当社商品」本来の使い方以外のご利用
 - (b) 「利用条件等」から外れたご利用
 - (c) 本ご承諾事項「ご利用にあたってのご注意」に反するご利用
 - (d) 「当社」以外による改造、修理による場合
 - (e) 「当社」以外の者によるソフトウェアプログラムによる場合
 - (f) 「当社」からの出荷時の科学・技術の水準では予見できなかった原因
 - (g) 上記のほか「当社」または「当社商品」以外の原因（天災等の不可抗力を含む）

● 責任の制限

本ご承諾事項に記載の保証が、「当社商品」に関する保証のすべてです。

「当社商品」に関連して生じた損害について、「当社」および「当社商品」の販売店は責任を負いません。

● 輸出管理

「当社商品」または技術資料を、輸出または非居住者に提供する場合は、安全保障貿易管理に関する日本および関係各国の法令・規制を遵守ください。お客様が法令・規則に違反する場合には、「当社商品」または技術資料をご提供できない場合があります。

オートメーションソフトウェア Sysmac Studio

● 保証内容

(1) 保証期間

本ソフトウェアの保証期間は、ご購入後1年といたします。

(2) 保証範囲

- a) 本ソフトウェアの使用許諾を受けたお客様が、上記保証期間中にコンピュータ・プログラムの瑕疵（『Sysmac Studio Version 1 オペレーションマニュアル（SBCA-470）』との重要な不一致）を発見し、当社に返却した場合は、当社は瑕疵のないコンピュータ・プログラムを記録した媒体と交換いたします。もしくは、当社の選択により、当社ホームページより瑕疵のないコンピュータ・プログラムをダウンロードしていただく方法により提供いたします。また、当社の責任によるコンピュータ・プログラムの記録媒体の不良を発見し、当社に返却した場合、当社は、無償で、良品の媒体に記録したコンピュータ・プログラムと交換いたします。
- b) 万一、当社がコンピュータ・プログラムの瑕疵を除去できないと判断した場合は、お客様が本ソフトウェア購入代金として支払った金額をお返しいたします。

● 責任の制限

- (1) 前条に定める交換または購入代金の返金は、本ソフトウェアの保証責任のすべてを定めるものであり、当社は本ソフトウェアの瑕疵により発生した、お客様の直接的、間接的あるいは波及効果による損害等いかなる損害に対しても一切の責任を負いません。
- (2) 当社は、本ソフトウェアを当社以外の第三者が変更、改造することにより生じた瑕疵につきましては、一切責任を負いません。
- (3) 当社は、本ソフトウェアに基づき、当社以外の第三者が開発したソフトウェアおよびそれにより生じた結果について一切の責任を負いません。

● 本ソフトウェアの用途

本ソフトウェアを『Sysmac Studio Version 1 オペレーションマニュアル（SBCA-470）』に記載の用途以外の用途で使用しないでください。

● 仕様の変更

本ソフトウェアの仕様および付属品は改善またはその他の事由により、必要に応じて、変更される場合があります。

● 適用範囲

以上の内容は、日本国内での取引および使用を前提としております。日本国外での取引および使用に関しては、当社営業担当者までご相談ください。

注意事項

- 実際のシステム構築に際しては、システムを構成する各機器や装置の仕様をご確認のうえ、定格や性能に対し余裕を持った使い方をしてください。万一故障があっても危険を最小にする安全回路などの安全対策を講じてください。
- システムを安全に使用していただくため、システムを構成する各機器や装置のマニュアルや取扱説明書などを入手してください。「安全上のご注意」「安全上の要点」「使用上の注意」など安全に関する注意事項を含め、内容を確認のうえ使用してください。
- システムが適合すべき規格や法規、または規制については、お客様自身で確認してください。

ソフトウェアのライセンスと著作権

本製品にはサードパーティ製のソフトウェアが組み込まれています。このソフトウェアに関連するライセンスと著作権については、http://www.fa.omron.co.jp/nj_info_j/ をご覧ください。

関連マニュアル

関連する NJ シリーズのマニュアルは、下表のとおりです。あわせてご覧ください。

マニュアル名称	Man.No.	形式	用途	内容
NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編	SBCA-466	形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ シリーズ CPU ユニットの概要／設計／取付／保守などの基本的な仕様について知りたいとき。 おもにハードウェアに関する情報。	NJ シリーズのシステム全体概要、および CPU ユニットに関して、以下の内容を説明します。 ・ 特長やシステム構成 ・ 概要 ・ 各部の名称と機能 ・ 一般仕様 ・ 設置と配線 ・ 保守点検
NJ/NX シリーズ CPU ユニット ユーザーズマニュアル ソフトウェア編	SBCA-467	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズ CPU ユニットのプログラミング／システムの立ち上げについて知りたいとき。 おもにソフトウェアに関する情報。	NJ/NX シリーズ CPU ユニットに関して、以下の内容を説明します。 ・ CPU ユニットの動作 ・ CPU ユニットの機能 ・ 初期設定 ・ IEC 61131-3 ベースの言語仕様とプログラミング
NJ/NX シリーズ コマンドリファレンス マニュアル 基本編	SBCA-468	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	オムロンが提供する命令仕様の詳細について知りたいとき。	各命令（IEC 61131-3 仕様）の詳細を説明します。
NJ/NX シリーズ CPU ユニット ユーザーズマニュアル モーション制御編	SBCE-433	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	モーション制御の設定やプログラミングの考え方について知りたいとき。	モーション制御のための CPU ユニットの設定や動作、プログラミングの考え方について説明します。
NJ/NX シリーズ コマンドリファレンス マニュアル モーション編	SBCE-434	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	モーション命令仕様の詳細について知りたいとき。	各モーション命令の詳細を説明します。
NJ/NX シリーズ CPU ユニット 内蔵 EtherCAT® ポート ユーザーズマニュアル	SBCD-376	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズ CPU ユニットの内蔵 EtherCAT ポートを使用するとき。	内蔵 EtherCAT ポートに関して説明します。 概要、構成、機能、セットアップについて記述しています。
NJ/NX シリーズ CPU ユニット 内蔵 EtherNet/IP™ ポート ユーザーズマニュアル	SBCD-377	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズ CPU ユニットの内蔵 EtherNet/IP ポートを使用するとき。	内蔵 EtherNet/IP ポートに関して説明します。 基本設定、タグデータリンク、その他の機能について記述しています。
NJ/NX シリーズ データベース接続 CPU ユニット ユーザーズマニュアル	SBCA-411	形 NX701-□□ 20 形 NX102-□□ 20 形 NJ501-□□ 20 形 NJ101-□□ 20	NJ/NX シリーズでデータベース接続サービス機能を使用するとき。	データベース接続サービス機能について説明します。
NJ/NX シリーズ トラブルシューティング マニュアル	SBCA-469	形 NX701-□□□□ 形 NX102-□□□□ 形 NX1P2-□□□□ 形 NJ501-□□□□ 形 NJ301-□□□□ 形 NJ101-□□□□	NJ/NX シリーズで検出する異常の詳細について知りたいとき。	NJ/NX シリーズ システムにて検出する異常管理の考え方と各異常項目について説明します。
Sysmac Studio Version 1 オペレーションマニュアル	SBCA-470	形 SYSMAC -SE2 □□□□	Sysmac Studio の操作方法、機能について知りたいとき	Sysmac Studio の操作方法について説明します。

マニュアル名称	Man.No.	形式	用途	内容
EtherCAT リモート I/O ターミナル GX シリーズ EtherCAT スレーブ ユーザーズマニュアル	SBCD-350	形 GX-ID □□□□ 形 GX-OD □□□□ 形 GX-OC □□□□ 形 GX-MD □□□□ 形 GX-AD □□□□ 形 GX-DA □□□□ 形 GX-EC □□□□ 形 XWT-ID □□ 形 XWT-OD □□	EtherCAT リモート I/O ターミナルの使用 方法について知りたいとき。	EtherCAT リモート I/O ターミナルの ハードウェアや設定方法、機能につい て説明します。
NJ シリーズ スタートアップガイド モーション制御編	SBCE-372	形 NJ501- □□□□ 形 NJ301- □□□□ 形 SYSMAC -SE2 □□□	NJ シリーズのモー ション制御機能を初 めて使用する場合の 立ち上げ手順と Sysmac Studio の操 作方法を知りたいと き。	軸のパラメータ設定、簡単な 1 軸位置 決めや 2 軸直線補間について、ハード ウェア組み立てからデバッグまでの操 作について説明します。

マニュアル改訂履歴

マニュアル改訂記号は、表表紙・裏表紙に記載されている Man.No. の後尾に付記されます。

Man.No.	SBCA-404C
---------	------------------

↑ 改訂記号

改訂記号	改訂年月	改訂理由・改訂ページ
A	2011 年 11 月	初版
B	2014 年 2 月	CPU ユニットおよび Sysmac Studio のバージョンアップに伴う改訂
C	2018 年 12 月	ご承諾事項の更新に伴う改訂、関連マニュアルの変更

NJ シリーズ CPU ユニットの特長

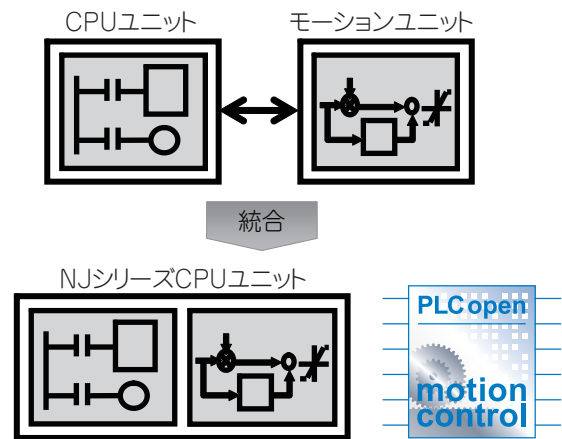
マシンオートメーションコントローラ NJ シリーズは、機械制御に必要な機能、高速性能と、産業用コントローラとしての安全性、信頼性、保守性とを両立した次世代コントローラです。プログラマブルコントローラの機能に、モーション制御に必要な機能を付加した統合型コントローラとして、視覚センサ、モーション、I/Oなどを、高速な EtherCAT 上で同期して制御できます。

高速かつ高精度な制御を実現

インテル®Atom™ プロセッサによって、ユーザプログラムを高速に実行できます。ラダー図命令は最速で 1.9ns、倍精度浮動小数点演算命令は最速で 26ns で実行できます。

シーケンス制御とモーション制御を統合

1 台の CPU ユニットでシーケンス制御とモーション制御の両方が実行できます。これにより、CPU ユニットと位置制御ユニット間の通信時間のロスがなくし、通信設定を容易にします。また、PLCopen に適合した豊富なモーションファンクションブロックにより、複雑なカム動作や多軸協調動作を可能にします。

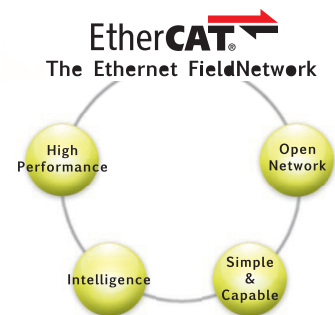


制御用ネットワーク EtherCAT を標準装備

CPU ユニットに EtherCAT 通信マスタ機能ポートを標準装備しています。

EtherCAT は、Ethernet システムをベースとし、より高速で高効率な通信を実現する高性能な産業用ネットワークシステムです。各ノードは Ethernet フレームを高速で転送するため、短い定周期な通信サイクルタイムを実現できます。

EtherCAT の標準採用により、機械制御に必要な、I/O システム、サーボドライバ、インバータ、マシンビジョンなどの機器をひとつのネットワークで接続できます。



EtherNet/IP 通信機能ポートを標準装備

CPU ユニットに EtherNet/IP 通信機能ポートを標準装備しています。

EtherNet/IP は、Ethernet を使用した産業用のマルチベンダネットワークです。コントローラ間のネットワークやフィールドネットワークとして使用できます。また、標準の Ethernet 技術が使用されているため、さまざまな汎用 Ethernet 機器を使用できます。

国際標準規格 IEC61131-3 準拠のプログラミング言語仕様

IEC61131-3 に準拠した言語仕様を搭載しています。一部、オムロン独自の改良を加えています。PLCopen に準拠したモーション制御命令をはじめ、IEC 規格に準拠した豊富な命令群（POU）が使用できます。

メモリマップを意識しない、変数によるプログラミング

パソコン上で高級言語の変数を使用するときと同様に、すべてのデータは変数によりアクセスします。変数を作成すると、CPU ユニットのメモリ上に変数が自動的に割り付けられるため、メモリマップを意識する必要はありません。このため、ハードウェア構成を変更しても、ソフトウェアの変更を最小限に抑えられます。

マルチタスク対応

I/O リフレッシュやユーザプログラム実行などの一連の処理をタスクに割り付け、タスクの実行条件と実行順序を指定することができます。複数のタスクを組み合わせることでアプリケーションに合わせて柔軟な制御が構築できます。

豊富なセキュリティ機能

操作権限の設定、ID によるユーザプログラムの実行制限など、豊富なセキュリティ機能が使用できます。

コントローラ全体の監視機能

装着されたユニット、EtherCAT スレーブを含めたコントローラ全体の事象（イベント）を、CPU ユニットで監視できます。

イベントとして異常が発生すると、対処方法が Sysmac Studio または NS シリーズ表示器でガイドされます。発生したイベントは履歴として記録されます。

オートメーションソフトウェア Sysmac Studio

Sysmac Studio は、コントローラだけでなく周辺機器や EtherCAT 上の機器にも対応した統合開発環境です。機器に関係ない、統一した操作ができます。設計から、デバッグ、シミュレーション、立ち上げ、運用開始後の変更までの、すべての工程の操作ができます。



豊富なシミュレーション機能

仮想コントローラ（シミュレータ）の実行機能、デバッグ機能、タスク実行時間の検証機能など、豊富なシミュレーション機能が使用できます。

目次

はじめに	1
対象となる読者の方々	1
対象となる製品	1
アイコン	1
ご承諾事項	2
注意事項	5
ソフトウェアのライセンスと著作権	5
関連マニュアル	6
マニュアル改訂履歴	8
NJ シリーズ CPU ユニットの特長	9

第 1 章 システム構成と立ち上げ手順

1-1 立ち上げ手順	1-2
1-2 システム構成と使用機器	1-3
1-3 作成するプログラム	1-4

第 2 章 プログラミングの基礎知識

2-1 POU（プログラム構成単位）	2-2
2-2 変数	2-3
2-3 タスク	2-4
2-4 プログラム言語	2-5
2-5 命令	2-6

第 3 章 事前に準備すること

3-1 Sysmac Studio をインストールする	3-2
3-2 ハードウェアを組み立てる	3-3
3-2-1 ユニットの装着する	3-3
3-2-2 デジタル I/O スレーブのノードアドレスを設定する	3-4
3-2-3 電源を配線する	3-4
3-2-4 EtherCAT 通信ケーブルを配線する	3-5
3-2-5 デジタル I/O スレーブとユニット電源を配線する	3-5

第 4 章 プログラミングとデバッグ

4-1 操作手順	4-2
4-2 新規プロジェクトを作成する	4-3
4-3 プログラミングする	4-6
4-3-1 グローバル変数を定義する	4-6
4-3-2 アルゴリズムを作成する	4-8

4-4	EtherCAT ネットワーク構成を作成する	4-18
4-5	デバイスの I/O 情報とプログラムをつなぐ	4-19
4-6	プログラムをデバッグする	4-20
4-6-1	オンラインデバッグの準備を行う	4-20
4-6-2	オフラインデバッグの準備を行う	4-26
4-6-3	ロジックをデバッグする	4-29
4-6-4	データトレースでランプの動作を確認する	4-38
4-6-5	オンラインエディットでロジックを変更する	4-42

第 5 章 便利な機能

5-1	アプリケーションのイベントを登録し、管理する	5-2
5-2	ユーザプログラム資産を守る	5-5
5-2-1	「ユーザプログラム実行用 ID 認証」による盗用防止	5-5
5-2-2	「ユーザプログラム復元情報なし転送」による盗用防止	5-6
5-2-3	「プロジェクトファイル全体のプロテクト機能」によるユーザ資産情報の保護	5-6
5-2-4	「データプロテクト機能」によるユーザ資産情報の保護	5-6
5-2-5	「操作権限認証」による誤操作の防止	5-7
5-2-6	「書込プロテクト機能」による Sysmac Studio からの書き込み操作禁止	5-8
5-2-7	「CPU ユニット名称機能」による Sysmac Studio の誤接続防止	5-8

付録

A-1	クロスリファレンスの操作方法	A-2
A-2	変数テーブル編集の便利な機能	A-4
A-3	よく使うプログラミング操作	A-6
A-4	オンラインのネットワーク構成の作成	A-13
A-5	オンラインデバッグとオフラインデバッグの違い	A-15
A-6	I/O マップでの強制値リフレッシュ	A-16

システム構成と立ち上げ手順

本章では、本ガイドで説明する立ち上げ手順と立ち上げに使用するシステム構成、および、作成するプログラムの動作を説明します。

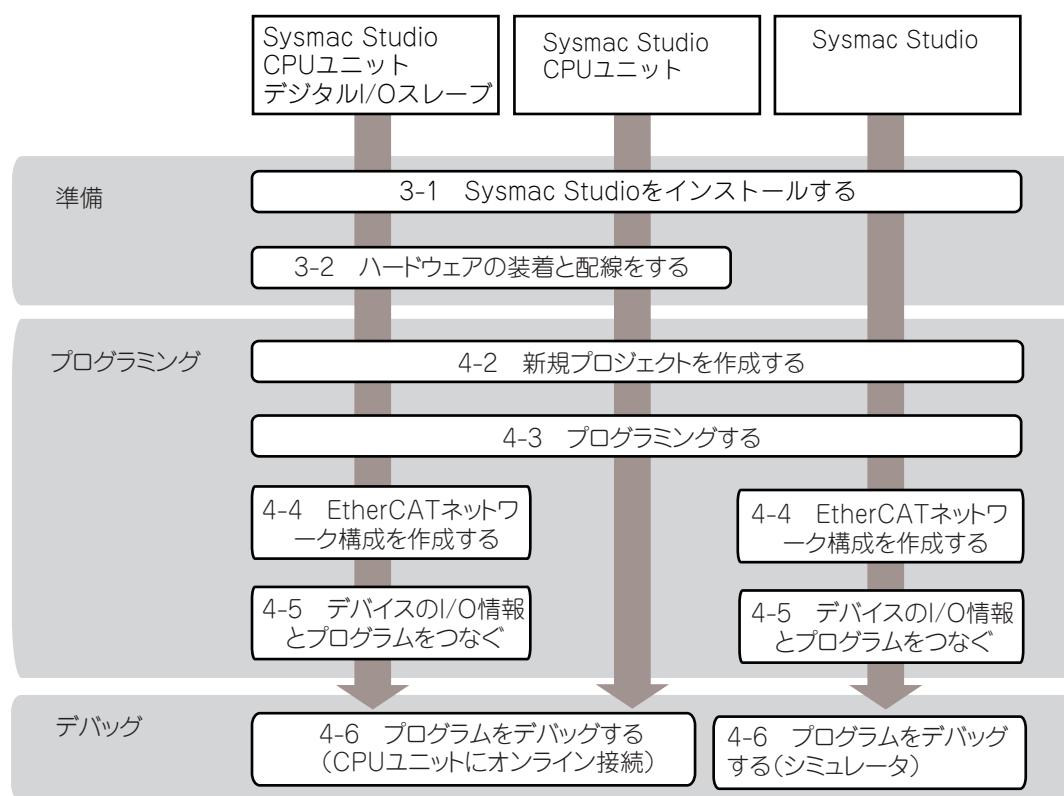
1-1 立ち上げ手順	1-2
1-2 システム構成と使用機器	1-3
1-3 作成するプログラム	1-4

1-1 立ち上げ手順

本ガイドでは、プログラミングからデバッグまでの操作を立ち上げ手順として説明します。プログラミングからデバッグまでの操作は、以下のシステム構成で行うことができます。

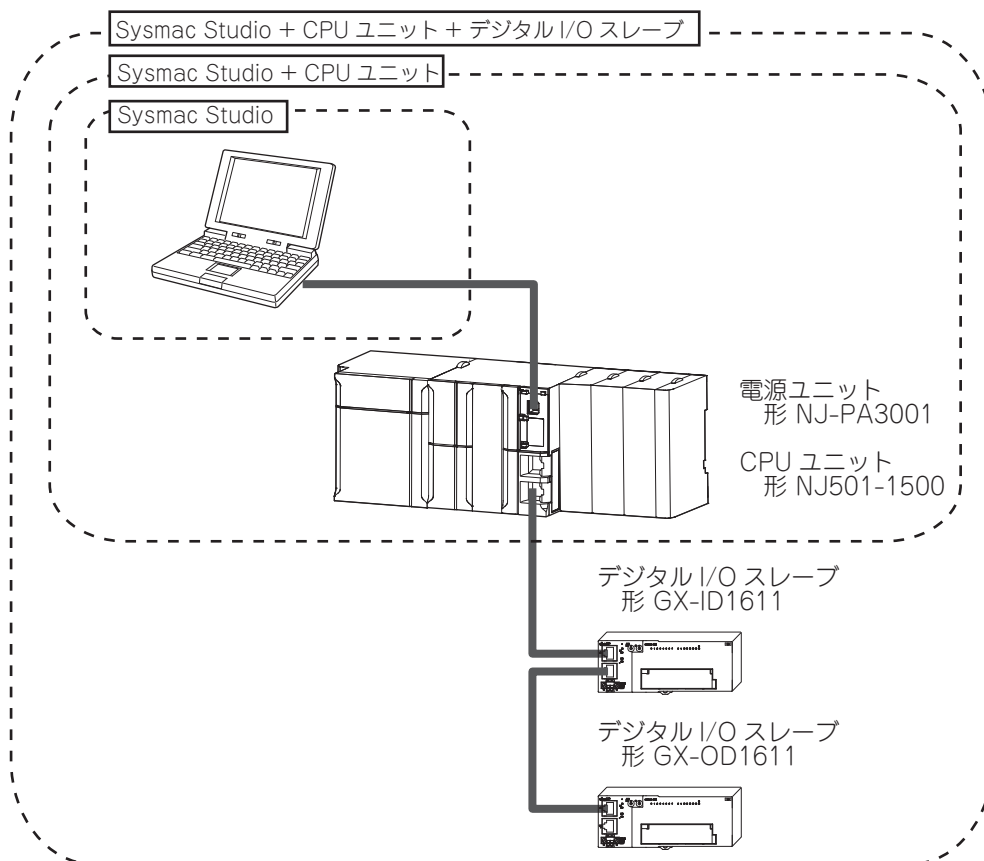
システム構成	操作内容
• Sysmac Studio • CPU ユニット • デジタル I/O スレーブ	Sysmac Studio でプログラミングし、CPU ユニットにオンライン接続してデバッグします。プログラミングでは、プログラムを作成し、デジタル I/O スレーブの I/O 情報をプログラムとつなぐ操作をします。
• Sysmac Studio • CPU ユニット	Sysmac Studio でプログラミングし、CPU ユニットにオンライン接続してデバッグします。デジタル I/O スレーブを使用しないため、プログラミングは、プログラムの作成だけです。
• Sysmac Studio	Sysmac Studio だけでプログラミングからデバッグまでを行います。プログラミングでは、プログラムを作成し、デジタル I/O スレーブの I/O 情報をプログラムとつなぐ操作をします。デバッグはシミュレータで行います。シミュレータでは、実機のシステム構成にあわせて作成したプログラムを、パソコン上でデバッグできます。

システム構成ごとの立ち上げ手順は下図のとおりです。



1-2 システム構成と使用機器

下図は本ガイドで使用するシステム構成です。CPU ユニットを使用する場合は、CPU ユニットのペリフェラル（USB）ポートに、Sysmac Studio をインストールしたパソコンを接続してください。デジタル I/O スレーブを使用する場合は、CPU ユニットの内蔵 EtherCAT ポートにデジタル I/O スレーブを接続してください。



● 使用機器

上記のシステム構成で使用する機器の形式は以下のとおりです。実際の使用機器を選定するには、各機器のマニュアルを参照してください。

機器名	形式	マニュアル名称
NJ シリーズ CPU ユニット	形 NJ501-1500 (Ver.1.00)	NJ シリーズ CPU ユニット ユーザーズマニュアル ハードウェア編 (SBCA-466)
NJ シリーズ電源ユニット	形 NJ-PA3001	
EtherCAT 通信ケーブル	形 XS5W-T421-CMD-K	GX シリーズ EtherCAT スレーブ ユーザーズマニュアル (SBCD-350)
デジタル I/O スレーブ	形 GX-ID1611(Ver.1.1)、 形 GX-OD1611(Ver.1.1)	
ユニット電源 ^{*1}	形 S8JX シリーズ [*]	—
USB ケーブル	市販 USB ケーブル ^{*2}	

^{*1}. デジタル I/O スレーブの電源として使用します。

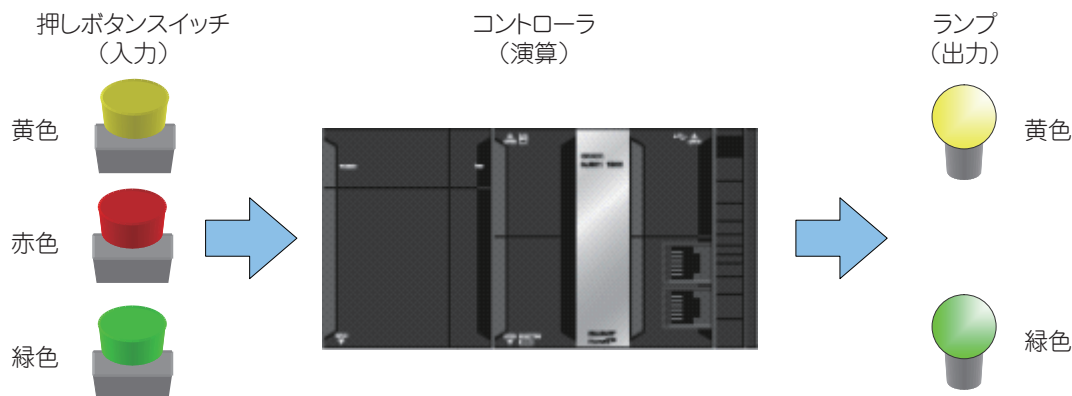
^{*2}. USB2.0（または 1.1）用ケーブル（A コネクタ -B コネクタ）最大 5.0m を使用します。

● オートメーションソフトウェア

品名	ライセンス数	形式
Sysmac Studio	なし（メディアのみ）	形 SYSMAC-SE200D
スタンダードエディション Ver.1.□□	1 ライセンス版	形 SYSMAC-SE201L

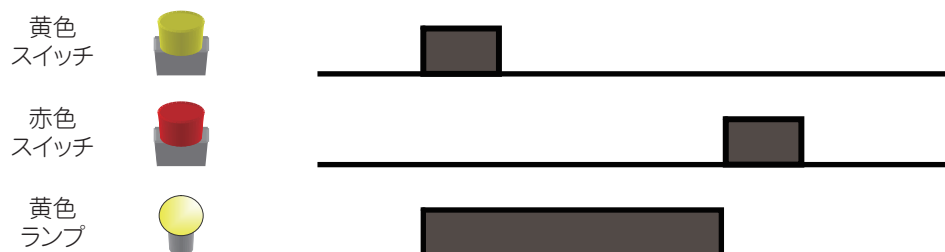
1-3 作成するプログラム

立ち上げ手順では、スイッチとランプを使用したシーケンス制御のプログラムを作成します。プログラムは、デジタル I/O スレーブの入力に押しボタンスイッチ、出力にランプを配線し、スイッチでランプを制御することを想定しています。ただし、実際にはプログラム上の変数で動作確認するため、スイッチとランプを準備する必要はありません。



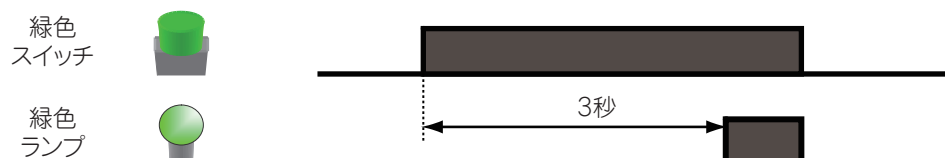
黄色のランプの制御（自己保持回路）

- 黄色の押しボタンスイッチを ON すると黄色のランプが点灯します。
- 黄色の押しボタンスイッチを OFF しても黄色のランプは消灯しません。
- 赤色の押しボタンスイッチを ON すると黄色のランプが消灯します。



緑色のランプの制御（オンディレイタイマ）

- 緑色の押しボタンスイッチを ON すると緑色のランプが 3 秒後に点灯します。
- 緑色の押しボタンスイッチを ON してから 3 秒以内に OFF した場合は、緑色のランプは点灯しません。
- 緑色のランプが点灯している状態で緑色の押しボタンスイッチを OFF すると、緑色のランプが消灯します。



2

プログラミングの基礎知識

本章では、NJ シリーズのプログラミングの基礎知識として必要な、「POU」、「変数」、「タスク」、「プログラミング言語」、「命令」について説明します。

2-1 POU（プログラム構成単位）	2-2
2-2 変数	2-3
2-3 タスク	2-4
2-4 プログラム言語	2-5
2-5 命令	2-6

2-1 POU（プログラム構成単位）

POU (Program Organization Unit) とは、IEC 61131-3 におけるユーザプログラムの実行単位です。複数の POU を組み合わせることにより、ユーザプログラム全体を構成します。

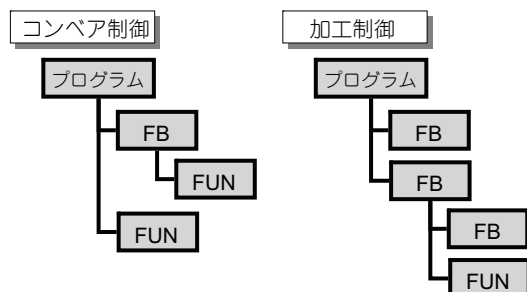
POU の構成要素には、以下の 3 種類があります。

POU の構成要素	説明
プログラム	メインルーチンに相当するアルゴリズムを記載する主となる要素です。アルゴリズムに、すべての命令、ファンクション、ファンクションブロックを記述することができます。
ファンクションブロック (以降「FB」と省略します)	入力の値が同じでも、異なる値の出力ができる要素です。プログラムまたは別の FB から呼び出されることで起動します。プログラム上で FB を使用するには、「インスタンス」としてプログラムに配置する必要があります。また、内部変数の値を保持することができます。そのため、タイマやカウンタのように状態を持つことができます。
ファンクション (以降「FUN」と省略します)	入力の値が同じであれば、常に均一の値を出力できる要素です。プログラム、FB、または FUN から呼び出されることで起動します。

● 視認性のよいプログラミング

ある POU から他の POU を呼び出すことにより、プログラムを階層的に構成することができます。たとえば、制御単位で階層化することで、プログラムの視認性を高めることができます。

下図はコンベア制御と加工制御で、それぞれ階層化した場合の例です。

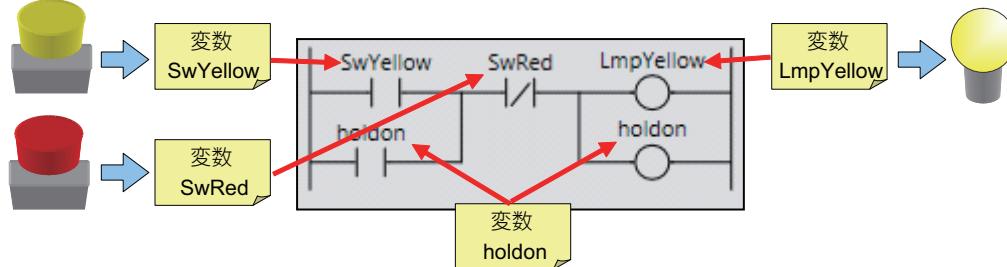


● 再利用性の高いプログラミング

FB および FUN は、プログラムを部品化するためのしくみです。ある処理を FB として部品化しておくと、同様な処理が必要な別の装置でも、FB のインスタンスを呼び出すことで再利用できます。

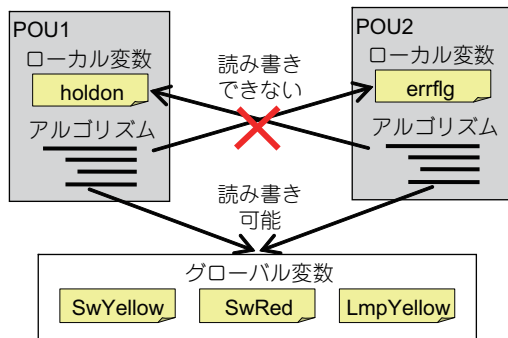
2-2 変数

変数とは、外部とやりとりした入出力データや、POU 内部処理時の一時データを格納する入れ物です。NJ シリーズでは、外部との入出力情報のやりとりやデータの演算などの処理を、すべて変数を介して行います。そのため、ハードウェアのメモリアドレスに依存しないプログラミングが可能です。



● ローカル変数とグローバル変数

ローカル変数は、定義された POU 内だけで読み書きが可能な変数です。また、グローバル変数は、すべての POU（プログラム、FB、FUN）から読み書きが可能な変数です。本ガイドのプログラムでは、押しボタンスイッチとランプにアクセスするための変数をグローバル変数として定義しています。また、自己保持回路で使用する変数はローカル変数として定義しています。



● 変数のデータ型

データ型とは、変数が表現する値の形式と範囲を特定する属性です。変数を定義する場合には、必ずデータ型を指定する必要があります。

NJ シリーズでは、あらかじめ仕様が決められている基本データ型として、以下の分類があります。他にもユーザが仕様を定義する派生データ型があります。

分類	定義	データ型
ブール型	値が 0 または 1 のいずれかのデータ型です。	BOOL
ビット列型	ビットの列で値を表現したデータ型です。	BYTE、WORD、DWORD、LWORD
整数型	値が整数値のデータ型です。	SINT、INT、DINT、LINT、USINT、UINT、UDINT、ULINT
実数型	値が実数値のデータ型です。	REAL、LREAL
持続時間型	値が時間（日・時・分・秒・ミリ秒）のデータ型です。	TIME
時刻型	値が時刻（時・分・秒）のデータ型です。	DATE
日付型	値が日付（年・月・日）のデータ型です。	TIME_OF_DAY
日付時刻型	値が日付時刻（年・月・日・時・分・秒・ミリ秒）のデータ型です。	DATE_AND_TIME
文字列型	値が文字列のデータ型です。	STRING

2-3 タスク

タスクとは、プログラムや I/O リフレッシュの実行条件や実行優先度を指定する単位です。タスクにプログラムを割り付けることで、プログラムを実行することができます。

タスクには、以下の 3 種類があります。

タスクの種類	定義
プライマリ定周期タスク	設定したタスク周期（タスクの実行間隔）に 1 回実行します。 タスクの実行優先度が最も高く、高速・高精度に実行することができるタスクです。
定周期タスク	設定したタスク周期に 1 回実行します。 プライマリ定周期タスクより実行優先度は低く、「プライマリ定周期タスク」の余り時間で実行されるタスクです。
イベントタスク ^{*1}	指定した実行条件が成立したときに 1 回だけ実行します。 プライマリ定周期タスクや、より実行優先度が高い定周期タスクが実行されている間は、実行を中断するタスクです。

^{*1}. イベントタスクは、CPU ユニット Ver.1.03 以降と Sysmac Studio Ver.1.04 以降との組み合わせで使用可能です。

本ガイドのプログラムでは、プライマリ定周期タスクでプログラムを実行しています。Sysmac Studio のデフォルト値である 1ms をタスク周期として使用するため、設定を変更する必要はありません。

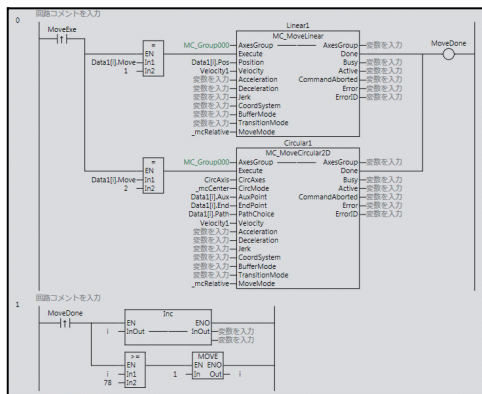
2-4 プログラム言語

POU のアルゴリズムを表現するための言語をプログラム言語と呼びます。

NJ シリーズで使用できるプログラム言語は、ラダー図言語 (LD)、ST 言語 (Structured Text : 構造化テキスト) の 2 種類です。

● ラダー図言語

ラダー図言語は、電気回路のイメージで記述したグラフィカルなプログラム言語です。FB や FUN を含めた、処理の各部品がすべて図で表されており、それらを直線で接続することにより、一連のアルゴリズムを構築します。ラダー図言語で書かれたアルゴリズムを、ラダー図と呼びます。



● ST 言語

ST 言語とは、IEC 61131-3 で規定されている、工業制御（おもにプログラマブルコントローラ）用の高級言語です。

用意された制御文、演算子、関数により、ラダー図言語では記述しにくい数値演算処理を部分的に処理するのに適しています。

```

1 // TableNo決定
2 FOR i:=0 TO ItemNum DO
3
4
5     IF (MinNo[i] <= ItemBox[i]) AND (ItemBox[i] <= MaxNo[i]) THEN // 範囲内
6         TableNo[i] := ItemBox[i];
7         RangeOK[i] := TRUE;
8
9     ELSIF (ItemBox[i] > MaxNo[i]) THEN // 上限超え
10        TableNo[i] := MaxNo[i];
11        RangeOK[i] := FALSE;
12
13     ELSE // 下限超え
14        TableNo[i] := MinNo[i];
15        RangeOK[i] := FALSE;
16
17     END_IF;
18
19 END_FOR;

```

本ガイドのプログラムでは、シーケンス制御に適している、ラダー図言語でアルゴリズムを作成します。

2-5 命令

命令とは、POU のアルゴリズムの中に記載する、オムロンが用意した最小単位の処理要素です。
命令には、以下の種類があります。

命令の種類	定 義
ラダー図構文（接点、コイル）	ラダー図言語の設定やコイルの形態をした命令です。 ラダー図言語で使用します。
FB 型命令	FB で作成された命令です。 タイマやカウンタのように、命令の内部で状態を保持する命令は、FB 型命令で作成されています。 プログラム上で FB 型命令を使用するには、「インスタンス」としてプログラムに配置する必要があります。
FUN 型命令	FUN で作成された命令です。 入力の値が同じであれば、常に均一の値の出力をする命令は、FUN 型命令で作成されています。
ST 言語の文	IF 文、FOR 文などように ST 言語の文の形態をした命令と、代入、演算子のように文の一部の形態をした命令があります。 ST 言語で使用します。

3

事前に準備すること

本章では、Sysmac Studio のインストール、およびハードウェアの装着と配線について説明します。

3-1	Sysmac Studio をインストールする	3-2
3-2	ハードウェアを組み立てる	3-3
3-2-1	ユニットを装着する	3-3
3-2-2	デジタル I/O スレーブのノードアドレスを設定する	3-4
3-2-3	電源を配線する	3-4
3-2-4	EtherCAT 通信ケーブルを配線する	3-5
3-2-5	デジタル I/O スレーブとユニット電源を配線する	3-5

3-1 Sysmac Studio をインストールする

NJ シリーズでは、コントローラ構成やコントローラ設定の作成、プログラム作成、デバッグ、シミュレーションを行うサポートソフトとして Sysmac Studio を使用します。

以下の手順で使用するパソコンに Sysmac Studio をインストールします。

- 1 Sysmac Studio のインストールディスクを DVD-ROM ドライブにセットしてください。**
セットアッププログラムが自動的に起動され、[セットアップ言語の選択] ダイアログボックスが表示されます。
- 2 インストールする言語を選択し、[OK] ボタンをクリックしてください。**
Sysmac Studio のセットアップウィザードが表示されます。
- 3 セットアップウィザードに従ってインストールしてください。**
- 4 インストールが完了したら、パソコンを再起動してください。**



参考

- Sysmac Studio をインストールするパソコンの動作環境は以下のとおりです。

OS		CPU	RAM	ディスプレイ
Windows XP SP3	最低	Celeron 540 (1.8GHz) 以上のプロセッサ	2GB	XGA
Windows Vista		を搭載した DOS/V (IBM AT 互換機)		1024 x 768
Windows 7 32		パーソナルコンピュータ		1600 万色
ビット版、64 ビット版	推奨	Core i5 M520 (2.4GHz)、または相当以上のプロセッサを搭載した DOS/V (IBM AT 互換機) パーソナルコンピュータ	2GB	WXGA
Windows 8 32				1280 x 800
ビット版、64 ビット版				1600 万色

- 上記の手順で Sysmac Studio がインストールできない場合は、『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』を参照してください。



使用上の注意

CX-One の Ver.4 より前のバージョンをインストールしている場合は、Sysmac Studio のインストールは中止され、インストールができません。この場合は、CX-One をアンインストールしてから Sysmac Studio をインストールしてください。

3-2 ハードウェアを組み立てる

システム構成の使用機器を接続し、配線します。

ここでは組み立て方法の概略を説明します。組み立て方法の詳細および安全に関する注意事項は、各機器のマニュアルを参照してください。



安全上の要点

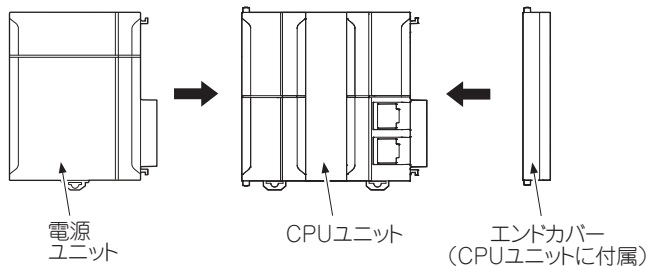
次のことを行うときは、コントローラの電源およびユニット電源を OFF にしてください。

- 各ユニットを着脱するとき
- ラックを組み立てるとき
- ディップスイッチやロータリスイッチを設定するとき
- ケーブルを接続、配線するとき
- コネクタを取り付けたり、取り外したりするとき

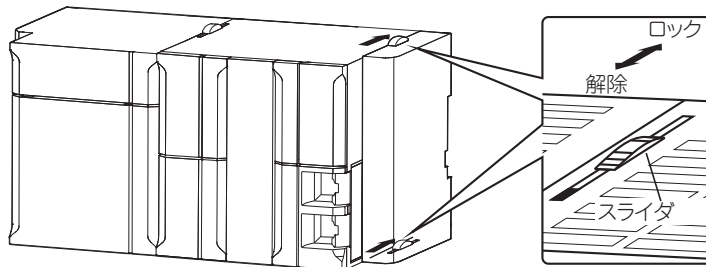
電源切断後、電源ユニットは数秒間コントローラに対して給電を継続する場合があります、この間 PWR LED が点灯します。上記の操作を行う場合は、PWR LED の消灯を確認してから行ってください。

3-2-1 ユニートを装着する

電源ユニットと CPU ユニット、およびエンドカバーを接続してください。

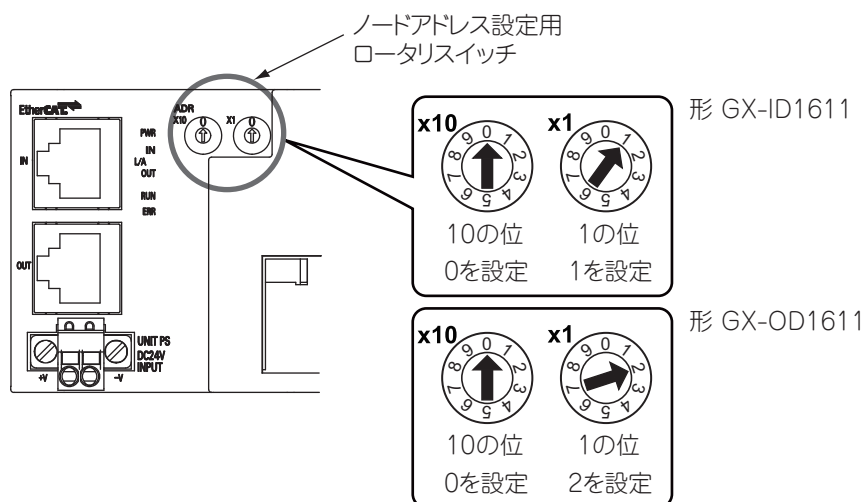


ユニット間コネクタを結合後、上下のスライダをカチッと音がするまでスライドさせて、確実にロックしてください。



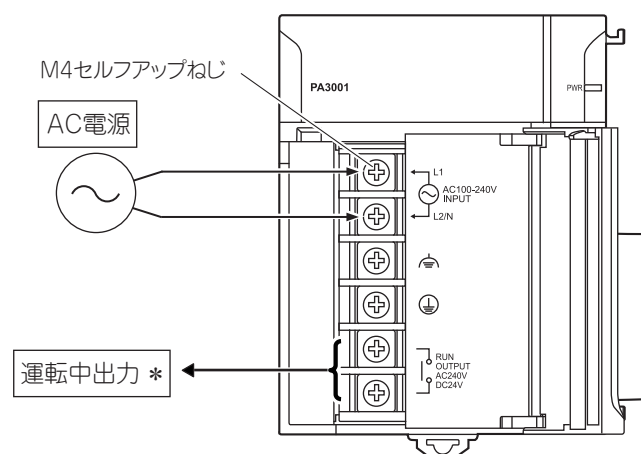
3-2-2 デジタル I/O スレーブのノードアドレスを設定する

デジタル I/O スレーブのノードアドレスを以下のように設定してください。



3-2-3 電源を配線する

電源ユニットの電源を配線してください。



* CPUユニットが「運転モード」のときON
「プログラムモード」または全停止フォールトレベルのコントローラ異常のときOFF

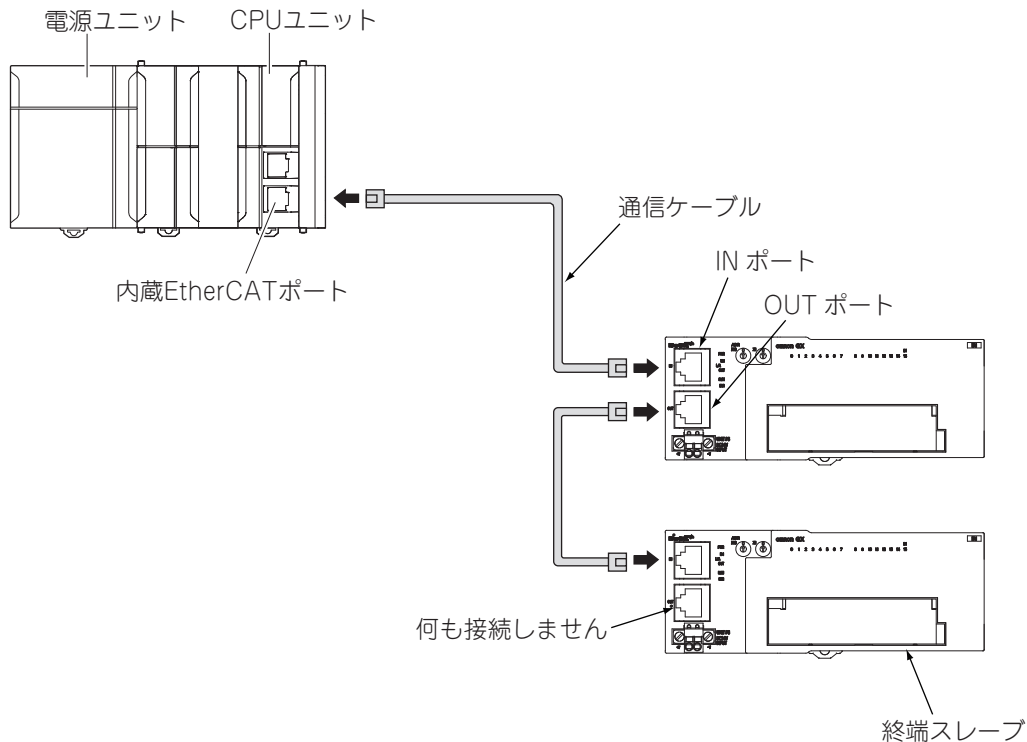


参考

本ガイドでは AC 電源タイプの電源ユニット形 NJ-PA3001 を使用していますが、DC 電源タイプの電源ユニット形 NJ-PD3001 も使用可能です。

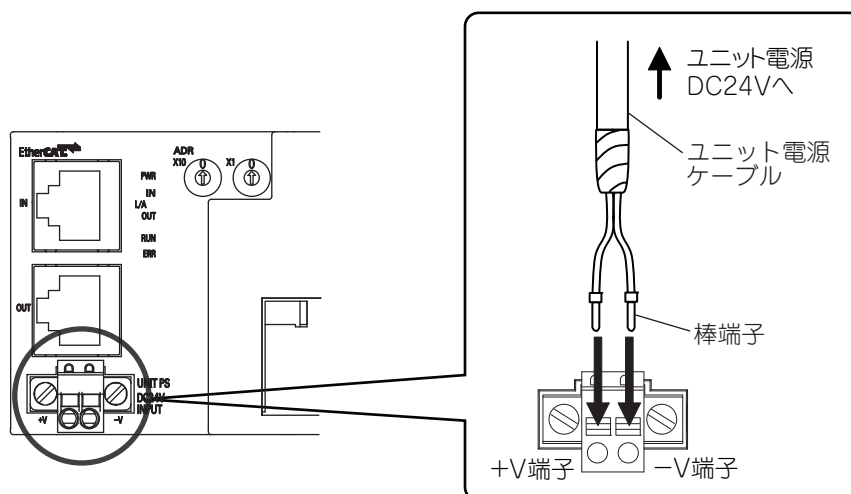
3-2-4 EtherCAT 通信ケーブルを配線する

内蔵 EtherCAT ポートと EtherCAT スレーブ通信ケーブルを下図のように配線してください。
内蔵 EtherCAT ポート側からの通信ケーブルをスレーブの IN ポートに、次のスレーブへの通信ケーブルを OUT ポートにそれぞれ接続してください。
なお、ネットワークの終端となるスレーブの OUT ポートには、何も接続しません。



3-2-5 デジタル I/O スレーブとユニット電源を配線する

各スレーブのユニット電源コネクタにユニット電源 (DC24V) からのケーブルを接続し、個別に給電してください。



4

プログラミングとデバッグ

本章では、プログラミングからデバッグまでの基本操作を説明します。

4

4-1	操作手順	4-2
4-2	新規プロジェクトを作成する	4-3
4-3	プログラミングする	4-6
4-3-1	グローバル変数を定義する	4-6
4-3-2	アルゴリズムを作成する	4-8
4-4	EtherCAT ネットワーク構成を作成する	4-18
4-5	デバイスの I/O 情報とプログラムをつなぐ	4-19
4-6	プログラムをデバッグする	4-20
4-6-1	オンラインデバッグの準備を行う	4-20
4-6-2	オフラインデバッグの準備を行う	4-26
4-6-3	ロジックをデバッグする	4-29
4-6-4	データトレースでランプの動作を確認する	4-38
4-6-5	オンラインエディットでロジックを変更する	4-42

4-1 操作手順

プログラミングからデバッグまでの基本的な操作手順を示します。

NJ シリーズでは、変数によるプログラミングができるため、アドレスによるメモリ空間を意識する必要がありません。これにより、ハードウェアとソフトウェアの設計を独立、並行して開発することが可能です。

本ガイドでは、変数によるプログラミングの特長を理解していただくために、プログラミングのときに実機がない場合の操作手順を説明します。

STEP 1. 新規プロジェクトを作成する (P.4-3)

プロジェクトファイルを作成します。



STEP 2. プログラミングする (P.4-6)

グローバル変数の定義、プログラムの作成を行います。

STEP2-1 グローバル変数を定義する (P.4-6)

STEP2-2 アルゴリズムを作成する (P.4-8)



STEP 3. EtherCAT ネットワーク構成を作成する (P.4-18)

CPU ユニットの内蔵 EtherCAT ポートに接続するデジタル I/O スレーブの構成を作成します。



STEP 4. デバイスの I/O 情報とプログラムをつなぐ (P.4-19)

デジタル I/O スレーブの I/O 情報にプログラムの変数を割り付けます。



STEP 5. プログラムをデバッグする (P.4-20)

CPU ユニットにプロジェクトを転送して動作確認（オンラインデバッグ）を行います。実機を使用しない場合は、Sysmac Studio のシミュレーション機能を使って動作確認（オフラインデバッグ）します。

STEP5-1 オンラインデバッグの準備を行う (P.4-20)

オフラインデバッグの準備を行う (P.4-26)

STEP5-2 ロジックをデバッグする (P.4-29)

STEP5-3 データトレースでランプの動作を確認する (P.4-38)

STEP5-4 オンラインエディットでロジックを変更する (P.4-42)

4-2 新規プロジェクトを作成する

Sysmac Studio を起動し、新規プロジェクトを作成します。

Sysmac Studio を起動する

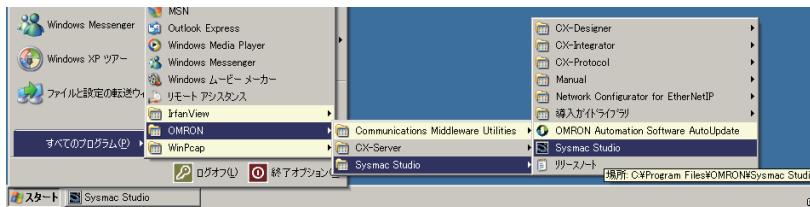
Sysmac Studio を起動します。

1 以下のいずれかの方法で Sysmac Studio を起動してください。

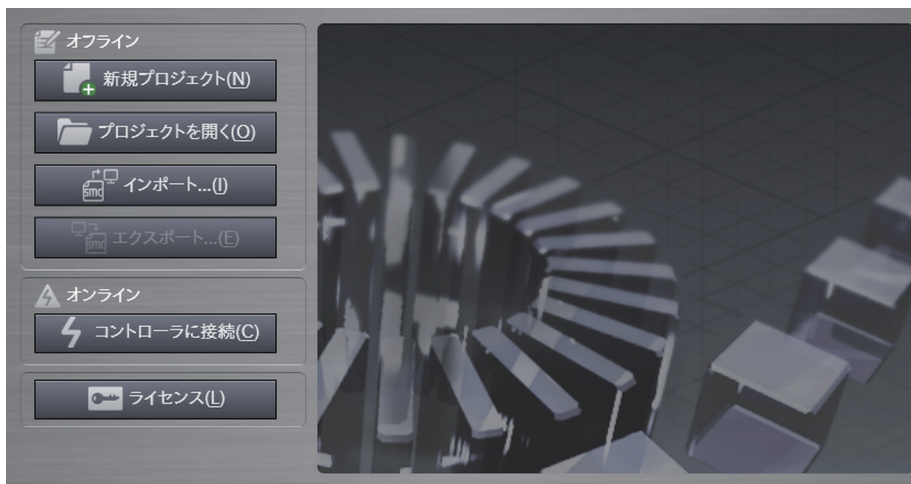
- デスクトップの [Sysmac Studio] のショートカットアイコンをダブルクリックしてください。



- Windows の [スタート] - [すべてのプログラム] - [OMRON] - [Sysmac Studio] から [Sysmac Studio] を選択してください。



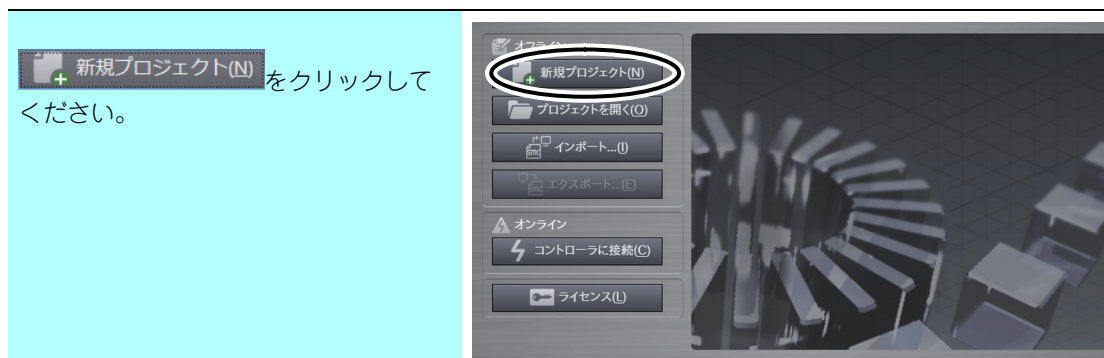
Sysmac Studio が起動し、以下の画面が表示されます。



新規プロジェクトを作成する

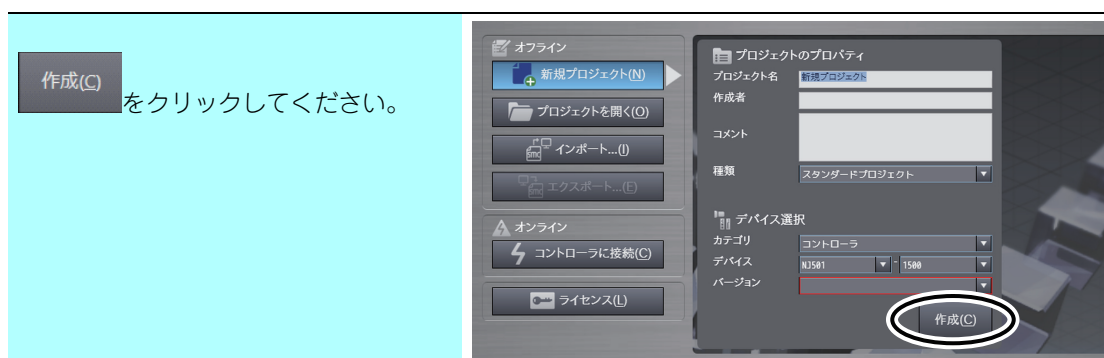
新規プロジェクトを作成します。

- 1 プロジェクトウィンドウで【新規プロジェクト】ボタンをクリックしてください。

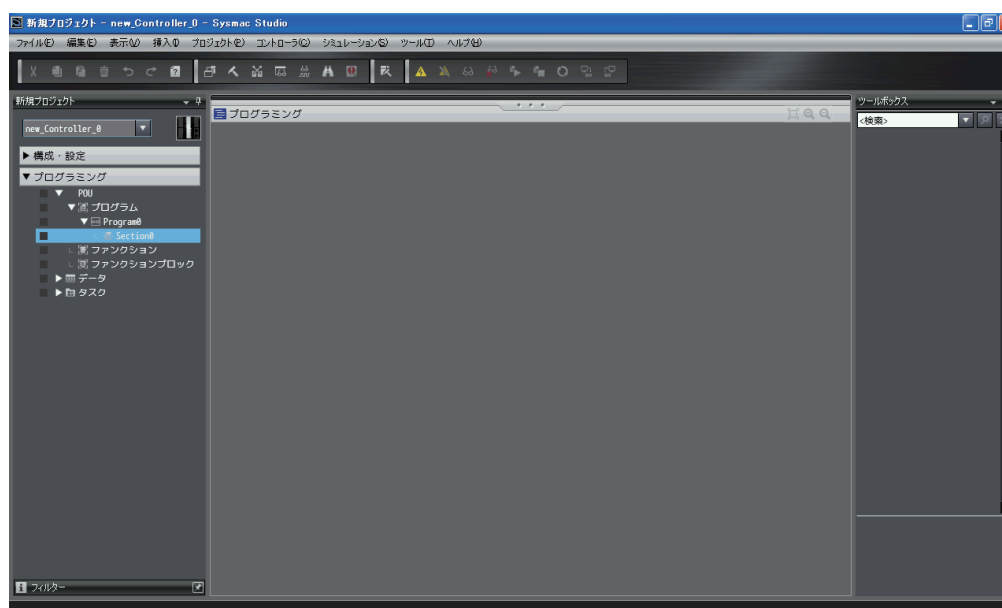


【プロジェクトのプロパティ】ダイアログボックスが表示されます。

- 2 【プロジェクトのプロパティ】ダイアログボックスで、【プロジェクト名】（必須）、【作成者】（任意）、【コメント】（任意）を入力し、デバイス選択の【カテゴリ】、【デバイス】から使用するデバイスを選択して、【作成】ボタンをクリックしてください。



プロジェクトファイルが作成され、次の画面が表示されます。

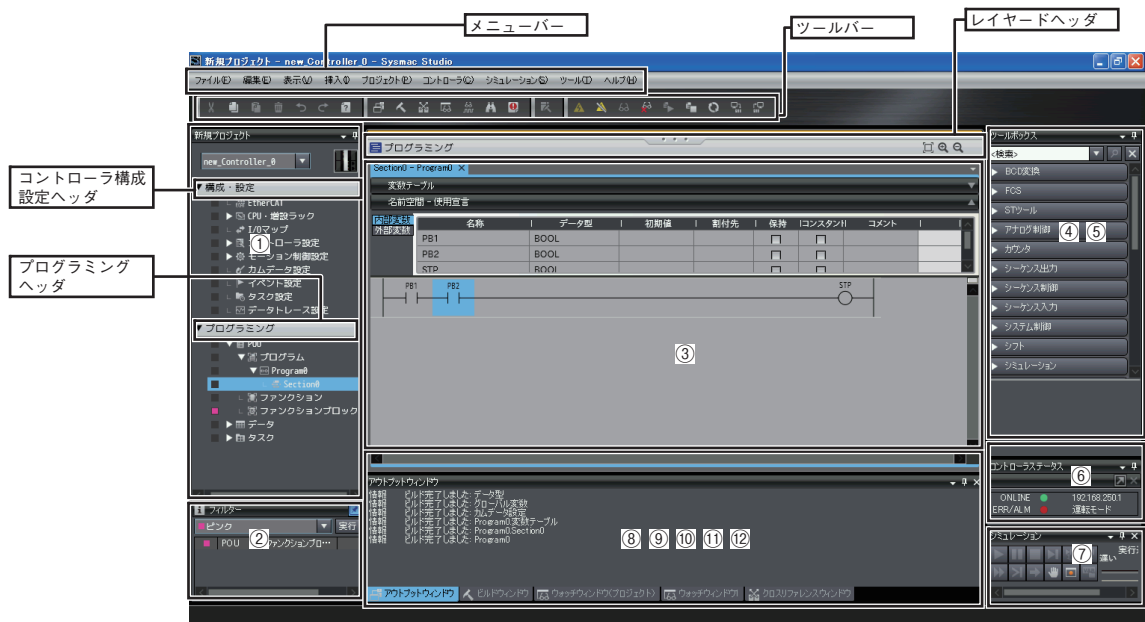


これで、プロジェクトファイルの新規作成が完了しました。



参考

以下に Sysmac Studio の画面各部の名称と機能を示します。



番号	名称	機能
①	マルチビューエクスプローラ	Sysmac Studio のすべてのデータにアクセスするための、アクセスポイントとなるウィンドウです。[構成・設定] と [プログラミング] のレイヤに分かれています。
②	フィルタウィンドウ	カラーコードや異常アイコンの表示されている項目を検索し、一覧表示することができるウィンドウです。
③	エディットウィンドウ	すべての項目の詳細データを表示/編集するウィンドウです。 [構成・設定] と [プログラミング] のレイヤ毎に 2 つのウィンドウがあります。
④	ツールボックス	エディットウィンドウに表示された詳細データの編集に使用する部品を表示するウィンドウです。
⑤	検索/置換ウィンドウ	プログラミングレイヤ内のデータを対象として、任意の文字列により検索・置換することができます。
⑥	コントローラステータスウィンドウ	コントローラの稼働状態などを表示します。コントローラに接続している状態だけで表示されます。
⑦	シミュレーションウィンドウ	コントローラのシミュレータの設定や起動/停止ができるウィンドウです。
⑧	クロスリファレンスウィンドウ	Sysmac Studio の変数、データ型、I/O ポート、ファンクション、ファンクションブロックの使用箇所を一覧表示するウィンドウです。
⑨	アウトプットウィンドウ	ビルドの結果が表示されるウィンドウです。
⑩	ウォッチウィンドウ	接続しているコントローラ、シミュレータのモニタ結果が表示されるウィンドウです。
⑪	ビルドウィンドウ	プログラムチェックおよびビルドの結果が表示されるウィンドウです。
⑫	検索/置換結果ウィンドウ	検索/置換機能で [すべて検索]、[すべて置換] を実行した結果が表示されるウィンドウです。

Sysmac Studio の各画面の詳細は、□『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』を参照してください。

4-3 プログラミングする

以下のプログラムを作成します。

● 黄色のランプの制御（自己保持回路）

- ・黄色の押しボタンスイッチを ON すると黄色のランプが点灯します
- ・黄色の押しボタンスイッチを OFF しても黄色のランプは消灯しません。
- ・赤色の押しボタンスイッチを ON すると黄色のランプが消灯します。

● 緑色のランプの制御（オンディレイタイマ）

- ・緑色の押しボタンスイッチを ON すると緑色のランプが3秒後に点灯します。
- ・緑色の押しボタンスイッチをONしてから3秒以内にOFFした場合は、緑色のランプは点灯しません。
- ・緑色のランプが点灯している状態で緑色の押しボタンスイッチを OFF すると、緑色のランプが消灯します。



参考

Sysmac Studio は、変数によるプログラミング環境を提供していますので、実機のシステム構成をプログラミングのときに意識する必要がありません。

4-3-1 グローバル変数を定義する

押しボタンスイッチとランプにアクセスするために必要なグローバル変数を定義します。定義するグローバル変数を下表に示します。

変数名	データ型	用途
SwYellow	BOOL	黄色の押しボタンスイッチ
SwRed	BOOL	赤色の押しボタンスイッチ
SwGreen	BOOL	緑色の押しボタンスイッチ
LmpYellow	BOOL	黄色のランプ
LmpGreen	BOOL	緑色のランプ

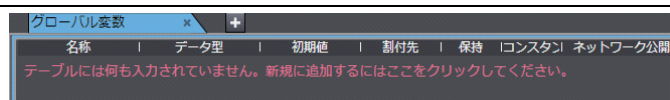
- 1 マルチビューエクスプローラの [プログラミング] - [データ] - [グローバル変数] をダブルクリックしてください。

[グローバル変数] をダブルクリックしてください。



- 2 グローバル変数テーブル内をクリックしてください。

グローバル変数テーブル内をクリックしてください。



変数 SwYellow を定義します。

3 名称欄に変数名を入力してください。

「SwYellow」と入力してください。

名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開
SwYellow	BOOL			☐	☐	非公開

 [Enter] キーで確定してください。

名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開
SwYellow	BOOL			☐	☐	非公開

データ型はデフォルトの BOOL 型を使用するため、データ型欄を変更する必要はありません。これで、変数 SwYellow の定義が完了しました。

次に、変数 SwRed を定義します。

4 グローバル変数テーブル内で [Insert] キーを押す、または、右クリックしてメニューから [新規作成] を選択してください。

 [Insert] キー、または、右クリックメニューから [新規作成] をクリックしてください。

名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開
SwYellow	BOOL			☐	☐	非公開

新規作成(N)
 切り取り(T)
 コピー(C)
 貼り付け(P)
 削除(D)
 元に戻す(U)
 やり直す(R)
 全て選択(A)

5 名称欄に変数名を入力してください。

「SwRed」と入力し、

 [Enter] キーで確定してください。

名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開
SwYellow	BOOL			☐	☐	非公開
SwRed	BOOL			☐	☐	非公開

同様に、変数 SwGreen、変数 LmpYellow、変数 LmpGreen を定義します。

6 同様に、変数 SwGreen、変数 LmpYellow、変数 LmpGreen を定義してください。

名称	データ型	初期値	割付先	保持	コンスタント	ネットワーク公開
SwYellow	BOOL			☐	☐	非公開
SwRed	BOOL			☐	☐	非公開
SwGreen	BOOL			☐	☐	非公開
LmpYellow	BOOL			☐	☐	非公開
LmpGreen	BOOL			☐	☐	非公開

これで、グローバル変数の定義が完了しました。



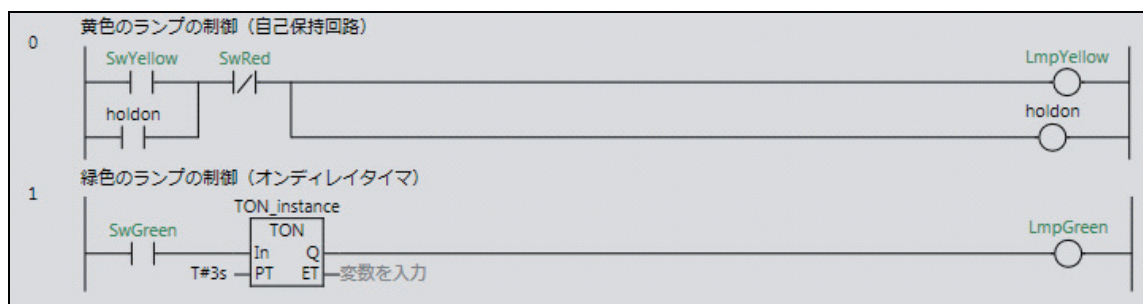
参考

- グローバル変数は、すべての POU（プログラム、FB、FUN）から読み書きが可能な変数です。グローバル変数をプロジェクトの複数箇所から読み書きしている場合は、どこで読み書きしているかをクロスリファレンス機能を使って確認することができます。クロスリファレンス機能の使い方は、□「A-1 クロスリファレンスの操作方法」(P.A-2)を参照してください。
- キー入力操作、名称入力アシスト機能、データ型入力アシスト機能により、変数の編集をスムーズに行うことができます。詳細は、□「A-2 変数テーブル編集の便利な機能」(P.A-4)を参照してください。

4-3-2 アルゴリズムを作成する

プログラムのアルゴリズムを作成します。アルゴリズムは、Program0 の Section0 に記述します。Program0 は、新規プロジェクト作成時に自動的に作成されるプログラムです。

以下のアルゴリズムを作成します。



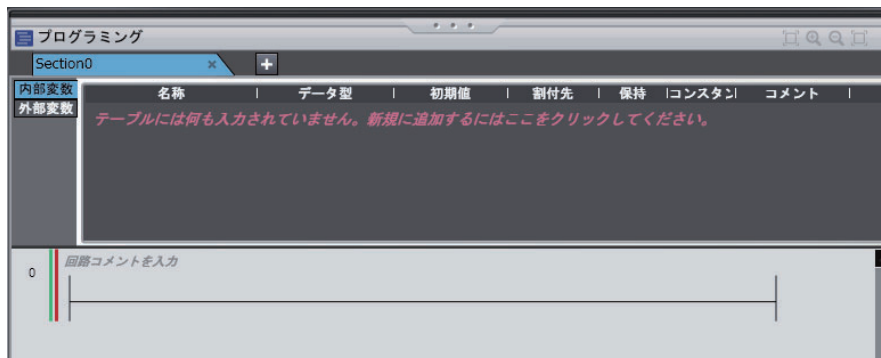
ラダーエディタを起動する

アルゴリズムを作成するためにラダーエディタを起動します。

- 1 マルチビューエクスプローラの [プログラミング] – [POU] – [プログラム] – [Program0] – [Section0] をダブルクリックしてください。



Section0 のラダーエディタが表示されます。



黄色のランプを制御するアルゴリズムを作成する（自己保持回路）

以下のアルゴリズムを作成します。

- 黄色の押しボタンスイッチを ON すると黄色のランプが点灯します。
- 黄色の押しボタンスイッチを OFF しても黄色のランプは消灯しません。
- 赤色を押しボタンスイッチを ON すると黄色のランプが消灯します。

黄色の押しボタンスイッチを OFF にしても黄色のランプは点灯のままとするので、自己保持回路を作成します。

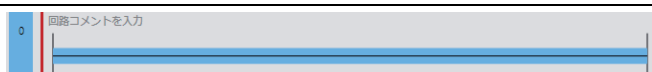
ここでは、ショートカットキーを使ってアルゴリズムを作成します。他にも、Sysmac Studio では、以下の方法でアルゴリズムを作成できます。

- ツールボックスから回路部品をドラッグ&ドロップ
- 接続線を選択して、右クリックメニューから「回路部品の挿入」を選択
- 接続線を選択して、メインメニューの「挿入」－「回路部品」から各回路部品を選択

1 接点を入力します。

変数 SwYellow を a 接点として入力します。

回路 0 の接続線をクリックしてください。

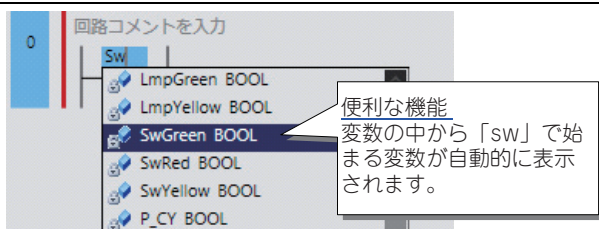


[C] キーで接点を挿入してください。



a 接点が挿入されます

「sw」と入力してください。



[↓] キーを 2 回押して [SwYellow BOOL] を選択し、
[Enter] キーを 2 回押して 確定してください。





参考

回路部品の入力時は、書式に誤りがあれば常にエラーとして表示されます。エラーがある場合は、回路番号と左母線との間に赤色の線が表示されます。



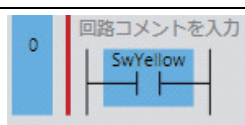
赤色の線にマウスのカーソルをあわせると、回路のエラーの内容が表示されます。



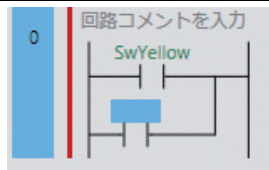
2 OR 回路を入力します。

変数 holdon を OR 回路で接続します。

変数 SwYellow の接点をクリックしてください。



[W] キーを押してください。



a 接点 (OR 回路) が挿入されます



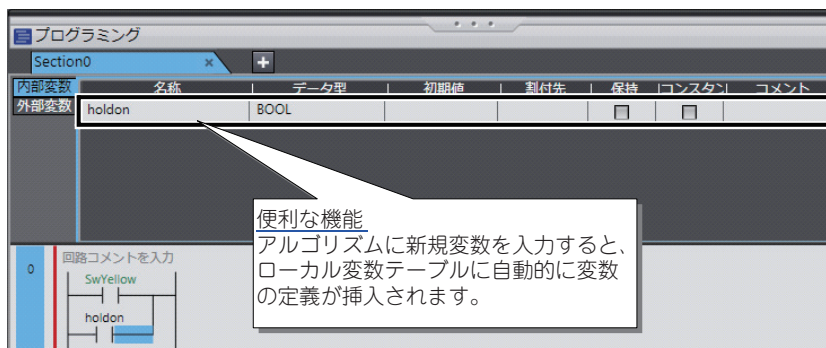
「holdon」と入力して、



[Enter] キーを2回押して確定してください。



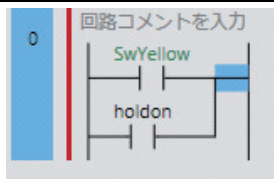
ラダーエディタで、holdon 変数を入力すると、Program0 のローカル変数テーブルに変数 holdon が自動的に挿入されます。



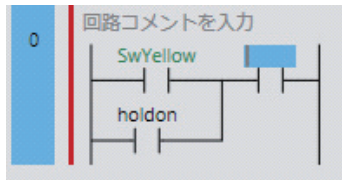
3 AND 回路を入力します。

変数 SwRed を b 接点として AND 回路で接続します。

AND 回路を挿入する接続線をクリックしてください。



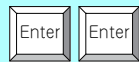
[C] キーを押してください。



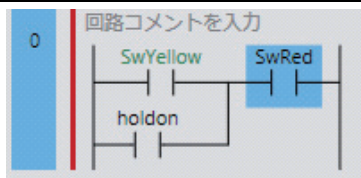
a 接点 (AND 回路) が挿入されます



「swred」と入力して、



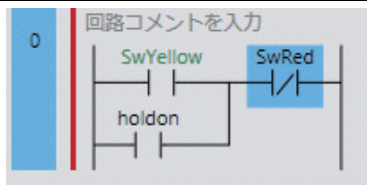
[Enter] キーを 2 回押して確定してください。



変数 SwRed の接点をクリックして、



[/] キーを押してください。

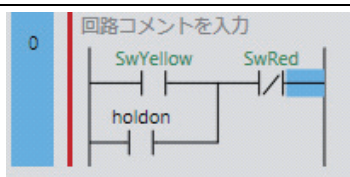


a 接点が b 接点に変更されます。

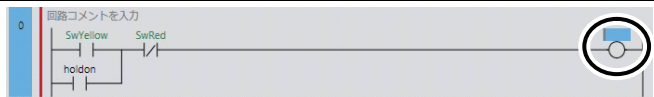
4 コイルを入力します。

変数 LmpYellow と変数 holdon をコイルとして接続します。

コイルを挿入する接続線をクリックしてください。



[O (オー)] キーでコイルを挿入してください。



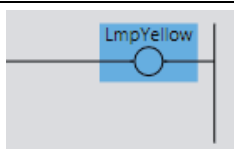
コイルが挿入されます



「LmpYellow」と入力して、



[Enter] キーを 2 回押して確定してください。



変数 LmpYellow のコイルをクリックして、



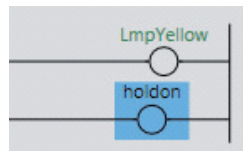
[O (オー)] キーでコイルを挿入してください。



「holdon」と入力して、



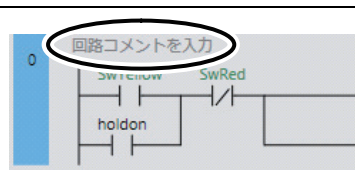
[Enter] キーを 2 回押して確定してください。



5 行コメントを入力します。

回路 0 にコメントを入力します。

「回路コメントを入力」をクリックしてください。



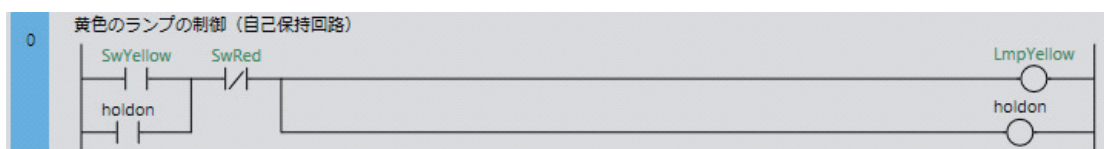
「黄色のランプの制御（自己保持回路）」と入力して、



[Enter] キーを 2 回押して確定してください。



これで、黄色のランプを制御するアルゴリズムの作成が完了しました。

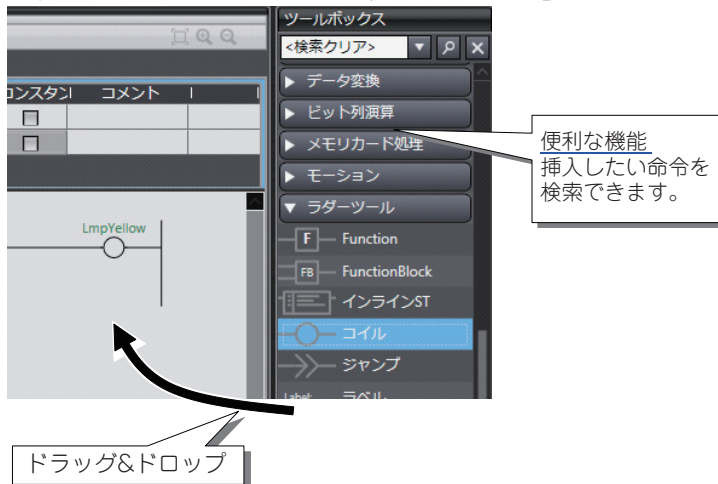




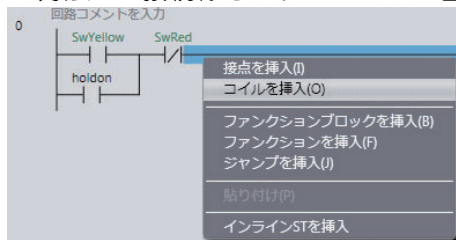
参考

ショートカットキー以外でのラダー回路の挿入方法は以下のとおりです。

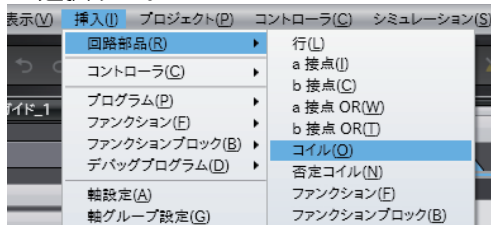
- 方法 1：ツールボックスの「ラダーツール」から回路部品をドラッグ&ドロップする。



- 方法 2：接続線をクリックして、右クリックメニューから回路部品を選択して挿入する。



- 方法 3：接続線をクリックして、メインメニューの「挿入」－「回路部品」から各回路部品を選択する。



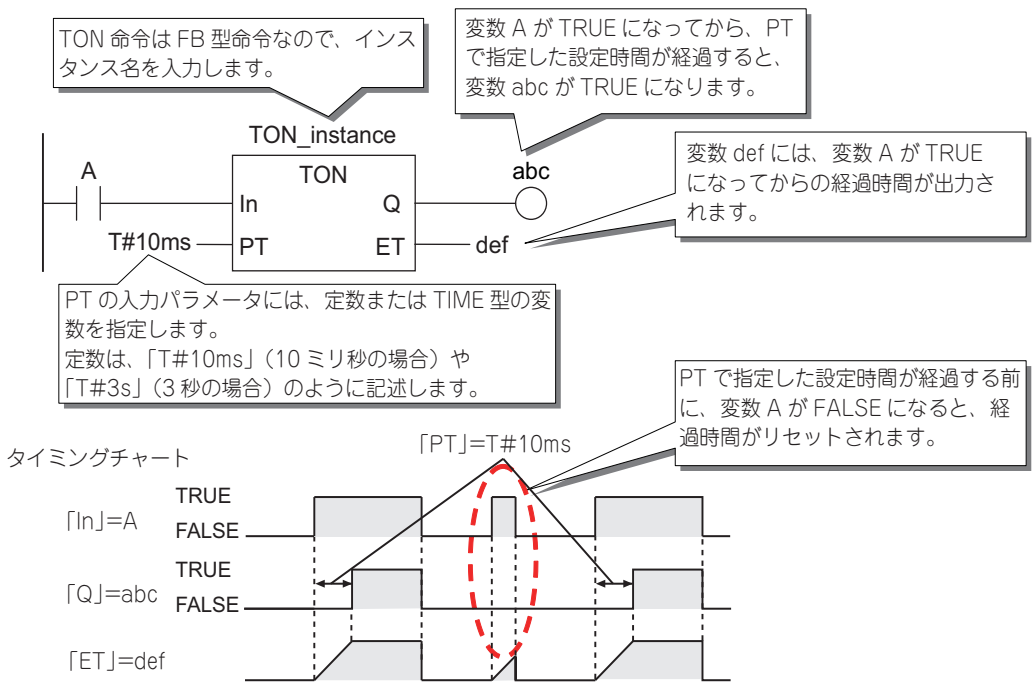
緑色のランプを制御するアルゴリズムを作成する（オンディレイタイマ）

以下のアルゴリズムを作成します。

- 緑色の押しボタンスイッチを ON すると緑色のランプが 3 秒後に点灯します。
- 緑色の押しボタンスイッチを ON してから 3 秒以内に OFF した場合は、緑色ランプは点灯しません。
- 緑色のランプが点灯している状態で緑色の押しボタンスイッチを OFF すると、緑色のランプが消灯します。

緑色の押しボタンスイッチを ON したときに、緑色のランプの点灯を遅れさせるには、オンディレイタイマ命令（TON）を使用します。

オンディレイタイマ命令（TON）は以下のように動作します。



オンディレイタイマ命令（TON）には以下の変数があります。

パラメータ	名称	入 / 出力	内容
In	タイマ入力	入力	TRUE : タイマ起動指示 FALSE : タイマリセット指示
PT	設定時間		タイマ起動から、「Q」が TRUE に変わるまでの時間
Q	タイマ出力	出力	TRUE : タイマ出力 ON FALSE : タイマ出力 OFF
ET	経過時間		タイマ起動からの経過時間



参考

NJ シリーズ CPU ユニットには、以下の 5 種類のタイマ命令があります。用途によって命令を使い分けことで、プログラムの視認性とプログラミングの生産性を高めることができます。

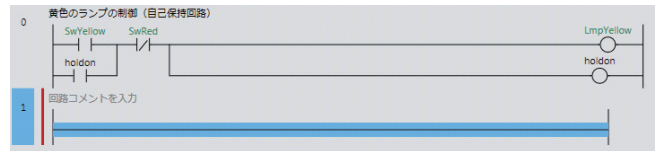
命 令	名 称	動 作
TON	オンディレイタイマ	起動から設定時間経過後に TRUE を出力するタイマです。
TOF	オフディレイタイマ	起動から設定時間経過後に FALSE を出力するタイマです。
TP	パルス出力	起動から設定時間だけ TRUE を出力するタイマです。
AccumulationTimer	積算タイマ	タイマ入力が TRUE である時間を積算するタイマです。
Timer	100ms タイマ	起動から設定時間経過後に TRUE を出力するタイマです。設定時間単位と計時精度は、ともに 100ms です。命令実行時間を短縮したい場合に使用します。

1 回路を挿入します。

回路 0 をクリックして選択してください。



 [R] キーを押してください。
または、右クリックメニューの [回路を挿入 (下)] を選択してください。*1



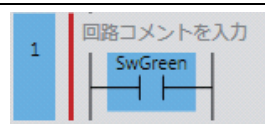
選択した回路の下に回路が挿入されます。

*1. 選択した回路の上に挿入する場合は、回路をクリックした後、[Shift] キーを押しながら [R] キーを押してください。または、右クリックメニューの [回路を挿入 (上)] を選択してください。

2 FB 型命令を挿入します。

オンディレイタイマ命令を挿入します。

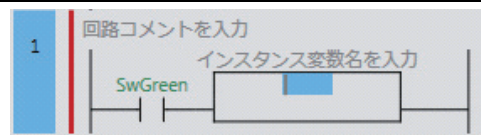
変数 SwGreen の a 接点を入力してください。



命令を挿入する位置の接続線をクリックしてください。

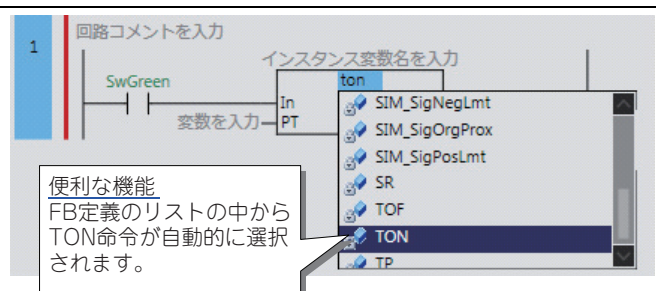


 [F] キーを押してください。*1

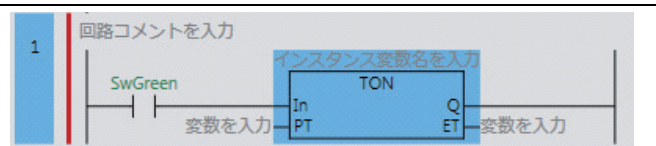


FB 定義名が未入力の FB が挿入されます。

「ton」と入力してください。



  [Enter] キーを 2 回押して、確定してください。

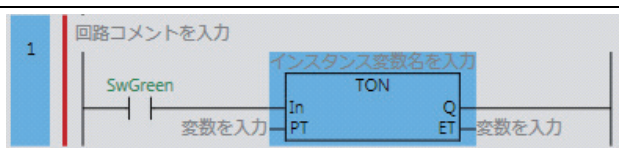


*1. FUN を入力したい場合は、[I (アイ)] キーを押してください。

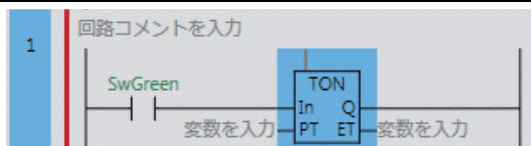
3 FB 型命令のインスタンス名を入力します。

挿入したオンディレイタイマ命令のインスタンス名を入力します。インスタンス名を「TON_instance」とします。

TON 命令をクリックしてください。



Enter [Enter] キーを押してください。
または、右クリックメニューから [インスタンス変数名を入力] をクリックしてください。



インスタンス名が入力できるようになります。

「TON_instance」と入力して、
Enter [Enter] キーで確定してください。

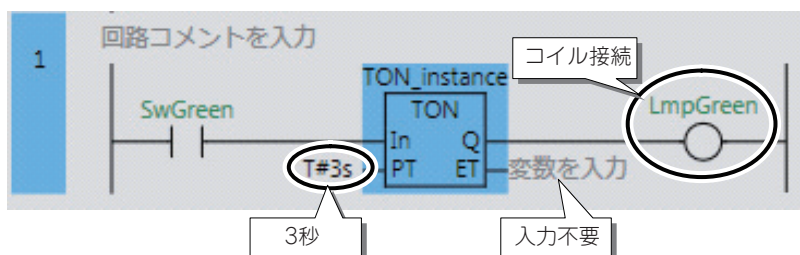


インスタンス名を入力すると、Program0 のローカル変数テーブルに変数 TON_instance が自動的に挿入されます。

プログラミング							
Section0							
内部変数	名称	データ型	初期値	割付先	保持	コンスタント	コメント
外部変数	holdon	BOOL			☐	☐	
	TON_instance	TON			☐	☐	

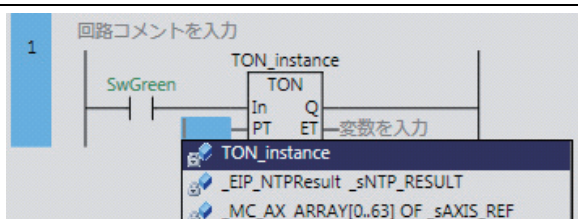
4 パラメータを入力します。

オンディレイタイマ命令のパラメータを入力します。



入力変数 PT の入力パラメータを入力します。3 秒後にランプを点灯させるので、'T#3s' と入力します。

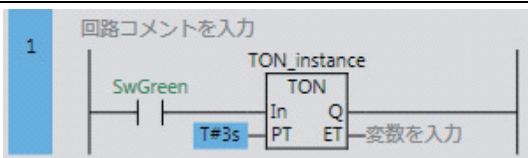
入力変数 PT の [変数を入力] をダブルクリックしてください。



入力パラメータが入力できるようになります。

「T#3s」と入力して、

 [Enter] キーで確定してください。

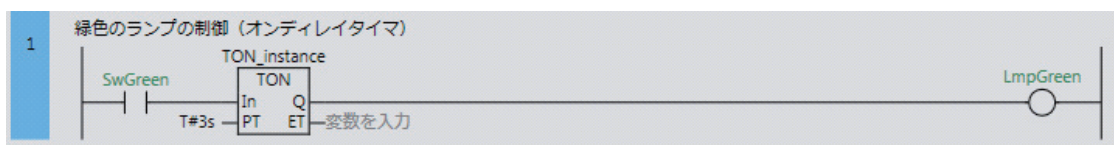


これで、入力変数 PT の入力パラメータの入力が完了しました。

このアルゴリズムでは、タイマ起動からの経過時間は必要ないため、出力変数 ET（経過時間）の入力は不要です。

出力変数 Q の右に変数 LmpGreen のコイルを接続すれば、オンディレイタイマ命令のパラメータの入力は完了です。

行コメントを入力して、緑色のランプを制御するアルゴリズムは完成です。



参考

微分型接点の入力や常時ON接点の入力など、よく使うプログラミング操作については、[□「A-3 よく使うプログラミング操作」](#)（P.A-6）を参照してください。

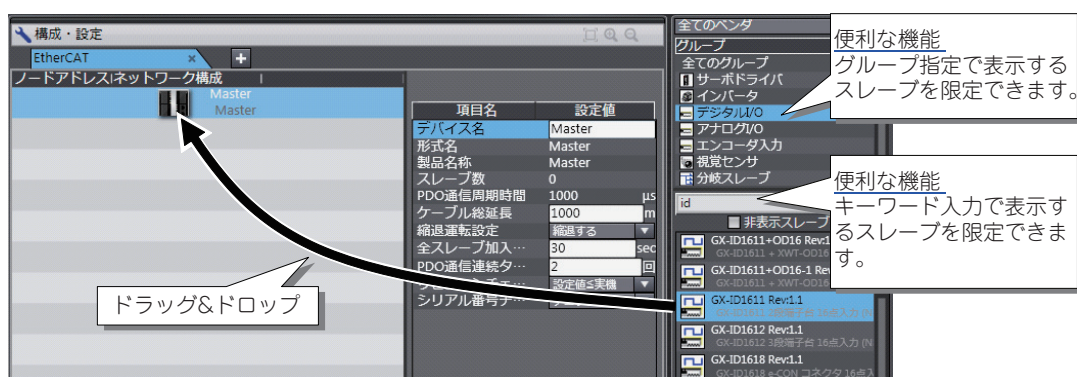
4-4 EtherCAT ネットワーク構成を作成する

EtherCATネットワーク構成に、デジタルI/Oスレーブ(形GX-ID1611、形GX-OD1611)を登録します。

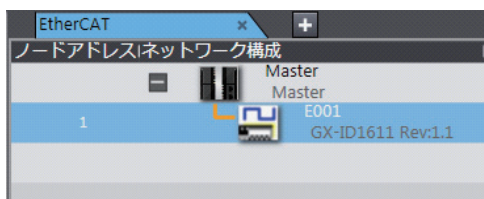
- 1 マルチビューエクスプローラから【構成・設定】－【EtherCAT】をダブルクリックしてください。または、【構成・設定】－【EtherCAT】を右クリックして【編集】を選択してください。



- 2 【ツールボックス】の「GX-ID1611 Rev:1.1」を、ネットワーク構成編集ウィンドウのマスタ上にドラッグ&ドロップしてください。または、ネットワーク構成編集ウィンドウでマスタを選択した状態で、【ツールボックス】の「GX-ID1611 Rev:1.1」をダブルクリックしてください。



マスタの下に GX-ID1611 が追加されます。



- 3 同様に、【ツールボックス】の「GX-OD1611 Rev:1.1」を、ネットワーク構成編集ウィンドウの GX-ID1611 にドラッグ&ドロップしてください。または、ネットワーク構成編集ウィンドウで GX-ID1611 を選択した状態で、【ツールボックス】の「GX-OD1611 Rev:1.1」をダブルクリックしてください。



これで、ネットワーク構成の作成が完了しました。



参考

EtherCAT の実ネットワーク構成の構築が完了している場合は、実ネットワーク構成から Sysmac Studio 上のネットワーク構成を自動構築することができます。詳細は、□「A-4 オンラインのネットワーク構成の作成」(P.A-13)を参照してください。

4-5 デバイスの I/O 情報とプログラムをつなぐ

EtherCAT スレーブや C.J ユニットなどのデバイス内のデータにアクセスする変数を「デバイス変数」と呼びます。

ここでは、押しボタンスイッチとランプにアクセスするために定義したグローバル変数を、デジタル I/O スレーブのデバイス変数に設定します。この操作により、デジタル I/O スレーブの I/O 情報とプログラムがつながります。

スレーブ位置	スレーブ名称	I/O ポート	説明	デバイス変数に設定する グローバル変数
ノード 1	GX-ID1611	In Bit00	入力接点 00	SwYellow
		In Bit01	入力接点 01	SwRed
		In Bit02	入力接点 02	SwGreen
ノード 2	GX-OD1611	Out Bit00	出力接点 00	LmpYellow
		Out Bit01	出力接点 01	LmpGreen

デバイス変数の設定は、I/O マップ画面で行います。

- 1 マルチビューエクスプローラの【構成・設定】－【I/O マップ】をダブルクリックしてください。または、【構成・設定】－【I/O マップ】を右クリックして【編集】を選択してください。



- 2 I/O マップで【Node1】－【GX-ID1611】－【入力接点 00】の変数の列をダブルクリックし、プルダウンリストからグローバル変数 SwYellow を選択してください。

位置	ポート	説明	R/W	データ型	変数
Node1	GX-ID1611	Read input 1st word	R	WORD	
Node2	GX-OD1611	Write output 1st word	W	WORD	

- 3 同様に、他のグローバル変数をデバイス変数に設定してください。

Node1	GX-ID1611	Read input 1st word	R	WORD	
Node2	GX-OD1611	Write output 1st word	W	WORD	

これで、デジタル I/O スレーブのデバイス変数の設定が完了しました。

4-6 プログラムをデバッグする

CPU ユニットにオンライン接続して動作確認することをオンラインデバッグと呼びます。CPU ユニットにオンライン接続せずに、パソコンの Sysmac Studio のシミュレータ上で動作確認することをオフラインデバッグと呼びます。実機がない場合は、オフラインデバッグで動作確認します。本ガイドの操作では、オンラインデバッグとオフラインデバッグでは、デバッグの操作は共通ですが、デバッグの準備内容が異なります。



参考

- オンラインデバッグとオフラインデバッグは、デバッグ機能の違いがあります。詳細は □「A-5 オンラインデバッグとオフラインデバッグの違い」(P.A-15) を参照してください。
- プログラムの作成時や変更時に、ビルドを行う必要があります。ビルドとは、プロジェクトのプログラムを CPU ユニットで実行できる形式に変換することです。ユーザ操作による変更時に Sysmac Studio が自動でビルドを実行するため、本ガイドではビルドの操作を省略しています。データ型、グローバル変数、プログラムのローカル変数およびアルゴリズムにユーザ操作による変更が入ってから 5 秒間ユーザ操作による変更がない場合、自動でビルドが開始されます。

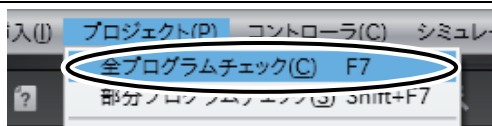
4-6-1 オンラインデバッグの準備を行う

オンラインデバッグの準備として、プログラムチェックと CPU ユニットへのプロジェクトの転送を行います。

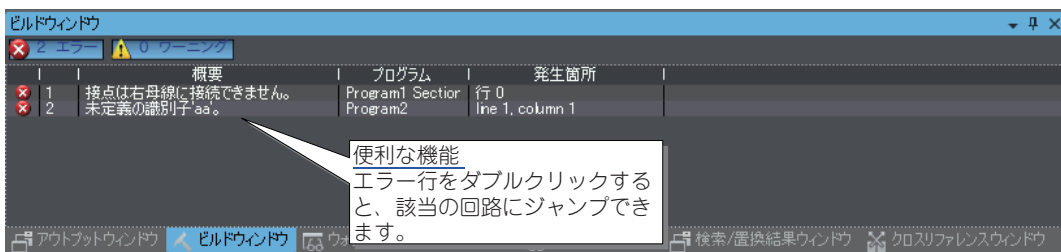
● プログラムチェックを実行する

1 全プログラムチェックを実行します。

メインメニューから [プロジェクト] - [全プログラムチェック] を選択してください。



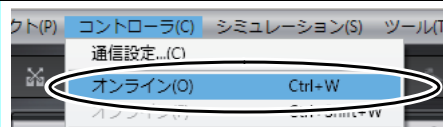
プログラムチェック結果がビルドウィンドウに表示されます。エラーがある場合はプログラムを修正してください。



● オンライン接続する

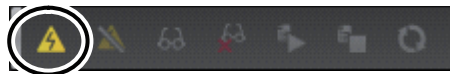
- 1 コントローラの電源を ON してください。
- 2 デジタルI/Oスレーブを接続している場合は、デジタルI/Oスレーブのユニット電源を ON してください。
- 3 以下のいずれかの方法で、オンライン接続してください。

方法 1：
メインメニューから [コントローラ] – [オンライン] を選択してください。



方法 2：

ツールバーの  をクリックしてください。

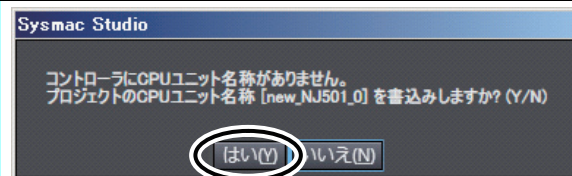


方法 3：

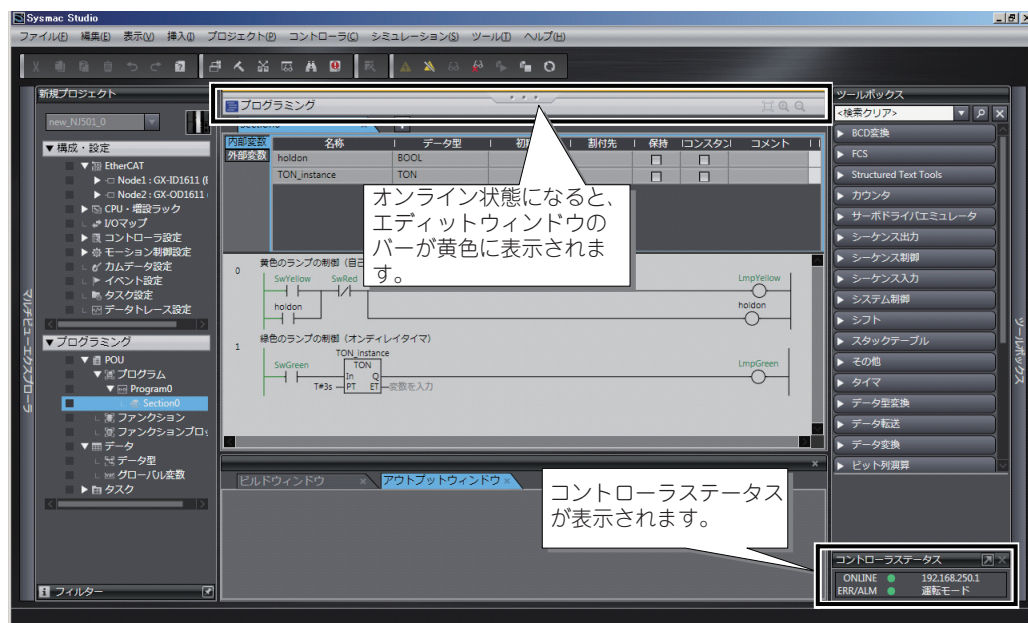
 [Ctrl] キー +  [W] キーを押してください。

- 4 コントローラに CPU ユニット名称が設定されていないときは、次のメッセージが表示されます。[はい] ボタンをクリックしてください。

 をクリックしてください。



オンライン接続状態となります。

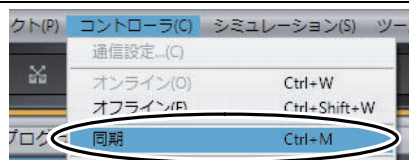


● プロジェクトを転送する

プロジェクトを CPU ユニットに転送します。プロジェクトの転送は、「同期」機能を使用します。Sysmac Studio の「同期」機能は、パソコン上のデータと実機コントローラのデータを自動照合し、どちらの方向に転送するかをユーザが選択する機能です。

1 以下のいずれかの方法で、同期ウィンドウを表示してください。

方法 1：
[コントローラ] - [同期] を選択してください。

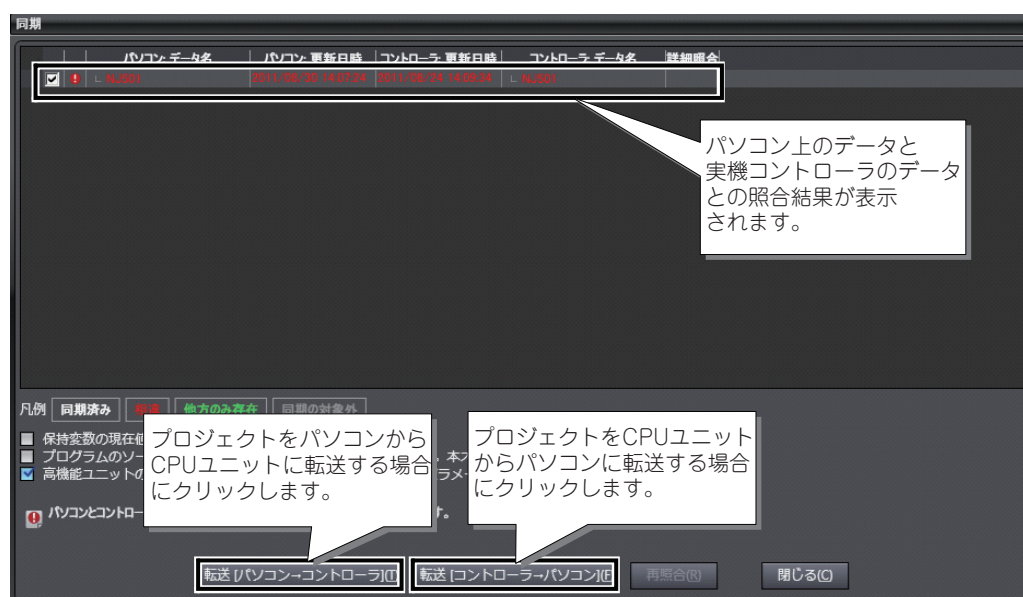


方法 2：
ツールバーの  をクリックしてください。



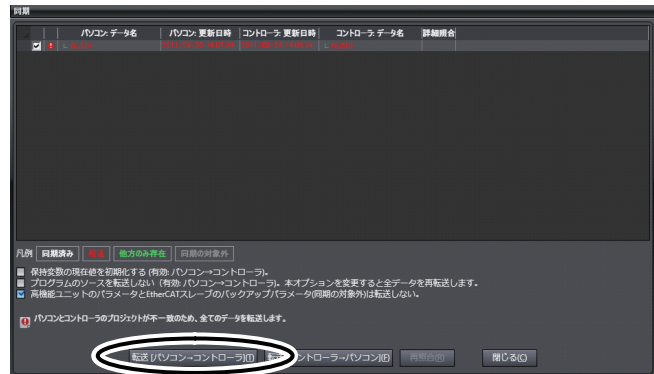
方法 3：
 [Ctrl] キー +  [M] キーを押してください。

パソコン上のデータと実機コントローラのデータの照合が開始されます。
照合が完了したら、照合結果が表示されます。



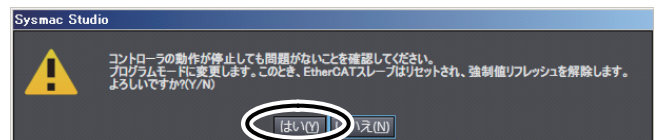
2 [転送 [パソコン→コントローラ]] ボタンをクリックしてください。

転送 [パソコン→コントローラ] をクリックしてください。

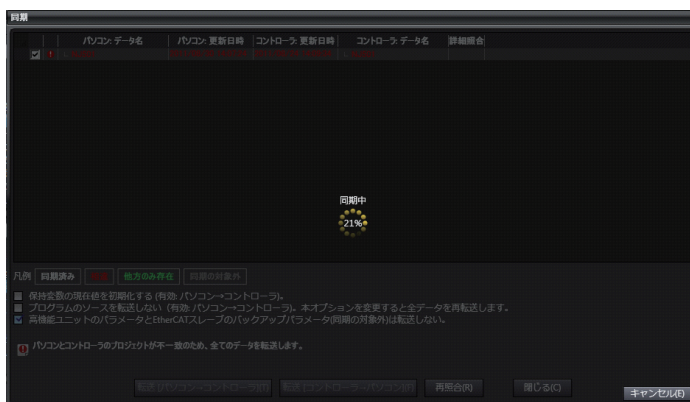


3 以下のダイアログが表示されますので、動作モードを変更しても問題がないことを確認して[はい] ボタンをクリックしてください。

はい(Y) をクリックしてください。

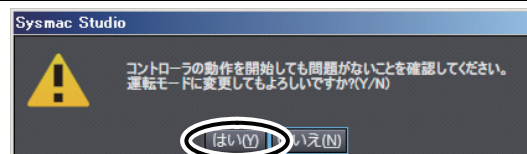


動作モードがプログラムモードに変更され、CPU ユニットへプロジェクトの転送が開始されます。転送中は、同期ウィンドウに進捗率が表示されます。



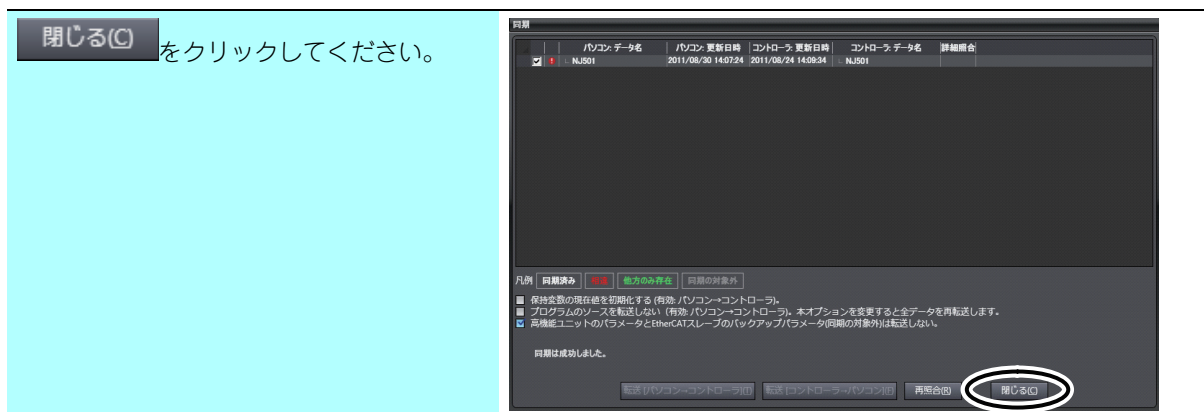
4 転送が完了したら、以下のダイアログが表示されますので、動作モードを変更しても問題がないことを確認して [はい] ボタンをクリックしてください。

はい(Y) をクリックしてください。



動作モードが運転モードに変更されます。

5 同期ウィンドウの右下の【閉じる】をクリックしてください。



同期ウィンドウが閉じられます。



使用上の注意

データ型、グローバル変数、プログラムのローカル変数およびアルゴリズムを変更すると、Sysmac Studio は自動的にビルドを行います。ビルド実行中は、CPU ユニットにプロジェクトを転送できません。ビルドが完了してからプロジェクトを転送してください。ビルドの実行状態は、ツール画面右下のプログレスバーに表示されます。

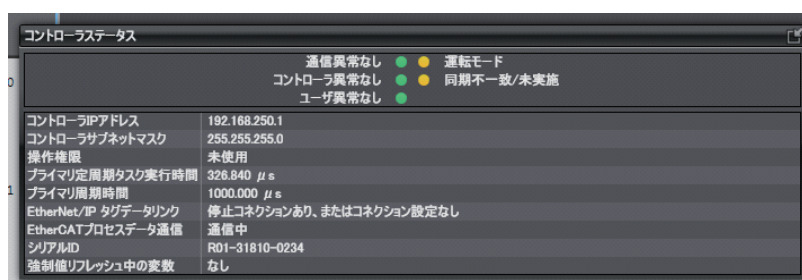


● コントローラの異常を確認する

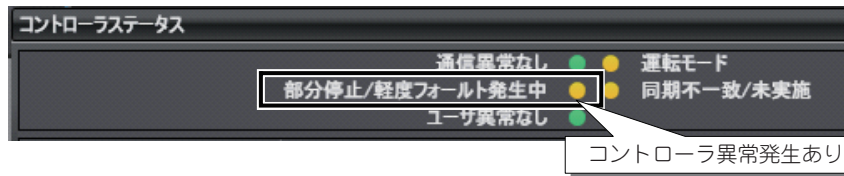
1 コントローラステータスの詳細画面ウィンドウを開きます。



コントローラステータスの詳細画面ウィンドウが表示されます



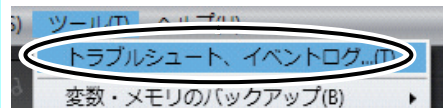
2 コントローラ異常が発生している場合は、[トラブルシュート、イベントログ] 画面を開きます。



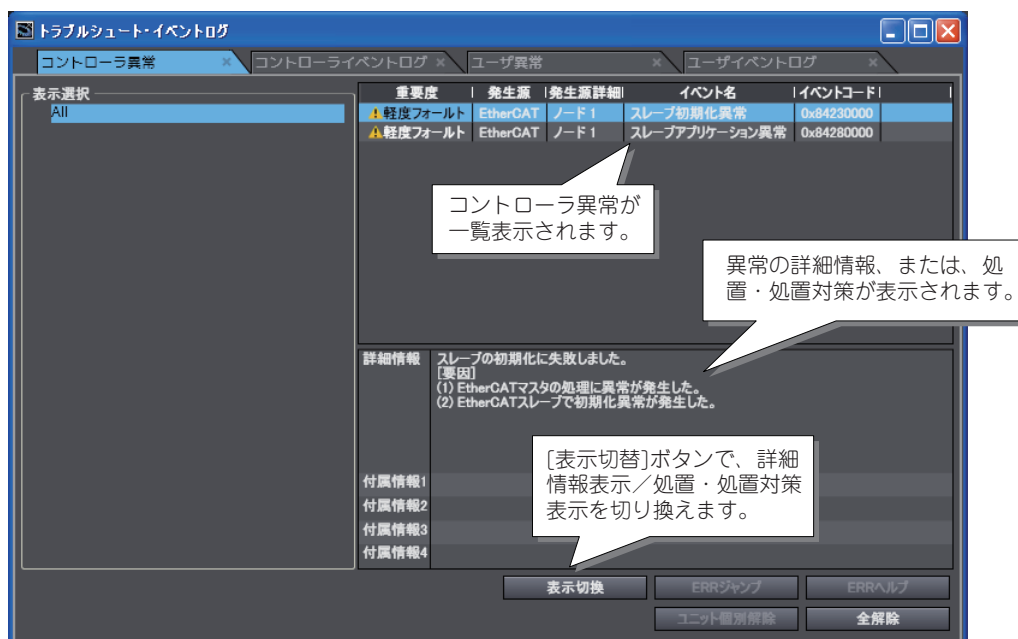
メインメニューから [ツール] - [トラブルシュート、イベントログ] を選択してください。

または、

ツールバーの  をクリックしてください。



トラブルシュート、イベントログ画面が表示されます。



3 異常の詳細情報とトラブルシューティング方法を参照し、異常の要因を取り除いてください。

4 [全解除] をクリックして、異常を解除してください。

 をクリックしてください。



発生している異常がすべて解除されます。

ただし、異常の要因が取り除かれていない場合は、再度異常が発生します。



使用上の注意

- EtherCAT 通信ケーブルが正しく接続されていない場合や、デジタル I/O スレーブのユニット電源が供給されていない場合は、軽度フォールトレベルのコントローラ異常（リンクオフ異常、または、ネットワーク構成照合異常）が発生します。EtherCAT 通信ケーブルが正しく接続されていること、デジタル I/O スレーブのユニット電源が供給されていることを確認してから異常を解除してください。
- 全停止フォールトレベルのコントローラ異常が発生している場合は、CPU ユニットの電源を再投入してください。

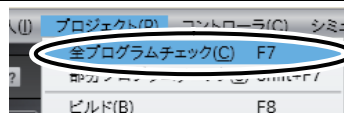
4-6-2 オフラインデバッグの準備を行う

オフラインデバッグの準備として、プログラムチェックとシミュレータの起動を行います。

● プログラムチェックを実行する

1 全プログラムチェックを実行します。

メインメニューから「プロジェクト」－「全プログラムチェック」を選択してください。

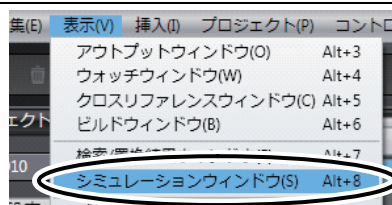


プログラムチェック結果がビルドウィンドウに表示されます。
エラーがある場合はプログラムを修正してください。

● シミュレータを起動する

1 以下のいずれかの方法で、シミュレーションウィンドウを表示します。

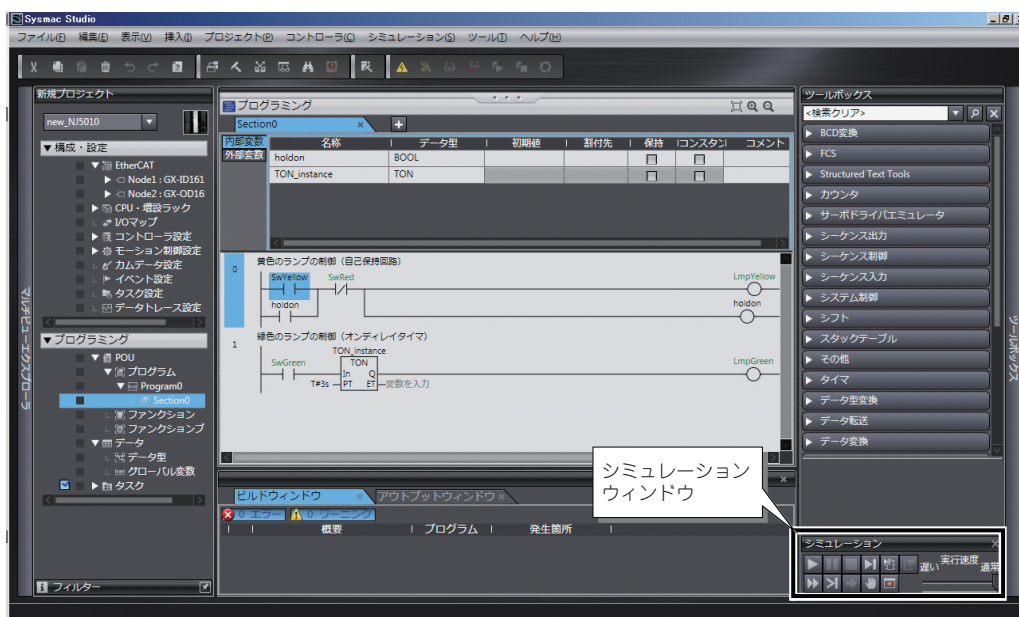
方法1：
メインメニューから「表示」－「シミュレーションウィンドウ」を選択してください。



方法2：

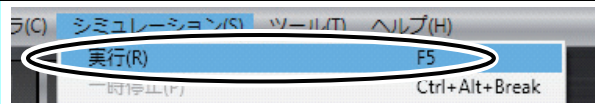
[Alt] キー + [8] キーを押してください。

シミュレーションウィンドウが表示されます。



2 以下のいずれかの方法で、シミュレータを起動します。

方法 1：
メインメニューから [シミュレーション]
－ [実行] を選択してください。

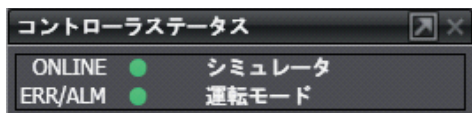


方法 2：
シミュレーションウィンドウの
 をクリックしてください。



方法 3：
 [F5] キーを押してください。

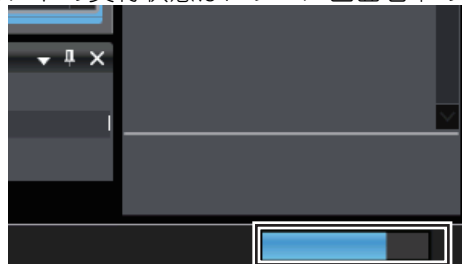
シミュレータの起動が完了すると、コントローラステータスウィンドウが以下のようになります。



これで、シミュレータの起動が完了しました。

使用上の注意

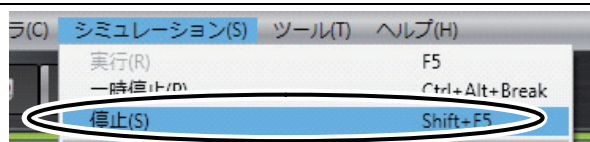
データ型、グローバル変数、プログラムのローカル変数およびアルゴリズムを変更すると、Sysmac Studio は自動的にプログラムのビルドを行います。ビルド実行中は、シミュレータを起動できません。ビルドが完了してからシミュレータを起動してください。
ビルドの実行状態は、ツール画面右下のプログレスバーに表示されます。



**参考**

- シミュレータは実際の時間より遅く動作します。このため、シミュレータでは、タイマ命令が起動して設定時間が経過するまでの時間が、CPU ユニットでの時間より遅くなります。
- シミュレーションを停止させる場合は、以下のいずれかの方法で行います。

方法 1：
メインメニューから [シミュレーション] - [停止] を選択してください。



方法 2：
シミュレーションウィンドウの
 をクリックしてください。



方法 3：
 [Shift] キー +  [F5]
キーを押してください。

4-6-3 ロジックをデバッグする

作成したプログラムのロジックが正しく動作するかをデバッグします。ここでは、黄色のランプを制御するアルゴリズムを例に、ロジックのデバッグ方法を説明します。



ロジックのデバッグでは、以下の機能を使用します。デジタル I/O スレーブを使用するシステム構成でオンラインデバッグする場合は、入力値を変更する方法として「強制値リフレッシュ」を使用します。オフラインデバッグする場合、および、デジタル I/O スレーブを使用しないシステム構成でオンラインデバッグする場合は、「セット／リセット」を使用します。

機能名	概要
モニタリング	コントローラの接点やコイルの ON/OFF 状態、および変数の現在値をモニタします。ラダーエディタ、ST エディタ、ウォッチウィンドウ、I/O マップで行うことができます。 ここでは、ラダーエディタとウォッチウィンドウでのモニタリングを使用します。
強制値リフレッシュ	デバッグ時などに、外部からの入力および外部への出力を、Sysmac Studio で指定した強制値でリフレッシュします。 デジタル I/O スレーブを使用するシステム構成でオンラインデバッグする場合に、押しボタンスイッチからの入力値の変更に使用します。
セット／リセット	任意の BOOL 型変数の状態を ON/OFF します。 オフラインデバッグする場合、および、デジタル I/O スレーブを使用しないシステム構成でオンラインデバッグする場合に、押しボタンスイッチからの入力値の変更に使用します。

参考

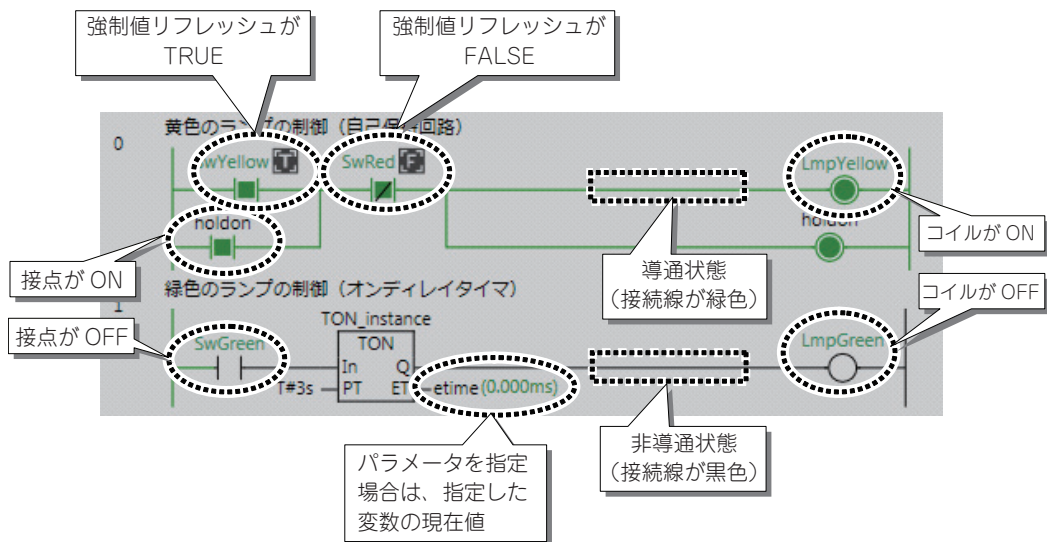
- スイッチなどを実際に接続していない入力スレーブは、入力が常に OFF になるため、「強制値リフレッシュ」を使用して動作確認します。「セット／リセット」では、変数の値を変更できません。EtherCAT スレーブや CJ ユニットを実際に接続してオンラインデバッグする場合は、「強制値リフレッシュ」を使って動作確認します。
- デバイス変数を [I/O マップ] で設定している場合は、シミュレータでも「強制値リフレッシュ」による動作確認ができます。ただし、シミュレータでは、スイッチなどの実際の入力に変数の値を書き換えることがないので、「セット／リセット」でも動作確認ができます。

モニタリング

ラダーエディタとウォッチウィンドウでのモニタリング方法を説明します。

● ラダーエディタでのモニタ

コントローラにオンライン接続した状態、または、シミュレータを起動した状態でラダーエディタを表示すると、ラダーエディタ上に回路がモニタ状態で表示されます。モニタ画面では、接点やコイルの ON/OFF、回路の導通、強制状態、パラメータに指定した変数の現在値などが確認できます。



● ウォッチウィンドウでのモニタ

複数の変数の現在値をウォッチウィンドウでまとめて確認することができます。

名称	モニタ値	変更	データ型	割付先	データ型式
SwYellow	False	TRUE FALSE	BOOL	ECAT://node#1/Read in	Boolean
Program0.etime	0.000ms		TIME		String
名前を入力...					

モニタ値
変数の現在値が表示されます。

変更
変更する値が表示されます。

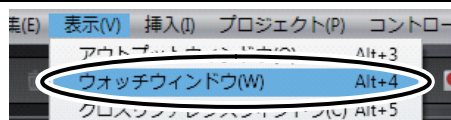
データ型
変数のデータ型が表示されます。

データ型式
現在値、変更する値の表示形式（10進、16進など）を選択できます。

変数 SwYellow を例にウォッチウィンドウへの変数の登録手順を説明します。

1 以下のいずれかの方法で、ウォッチウィンドウを表示します。

方法 1：
メインメニューから「表示」－「ウォッチウィンドウ」を選択してください。



方法 2：
[F6] をクリックしてください。



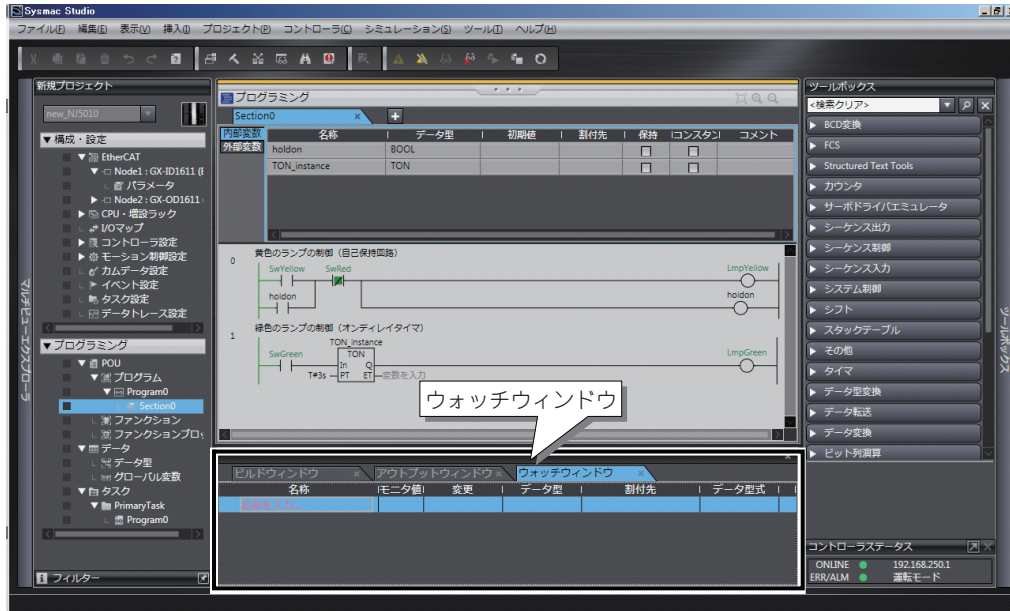
方法 3 :



[Alt] キー +

[4] キーを押し
てください。

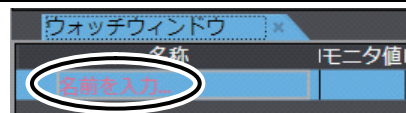
Sysmac Studio の画面下部にウォッチウィンドウが表示されます。



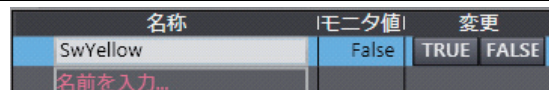
2 以下のいずれかの方法で、ウォッチウィンドウに変数を登録します。

方法 1 : 変数名入力

[名前を入力...] をクリックしてください。



「Sw」と入力してください。

便利な機能
変数の候補がリスト
に表示されます。[↓] キーを 2 回押して
「SwYellow」を選択し、[Enter] キーを押して確定してくだ
さい。

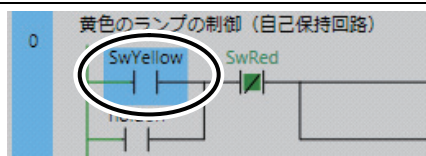
変数 SwYellow がウォッチウィンドウに登録されます。

他の変数を登録する場合の入力例は以下のとおりです。

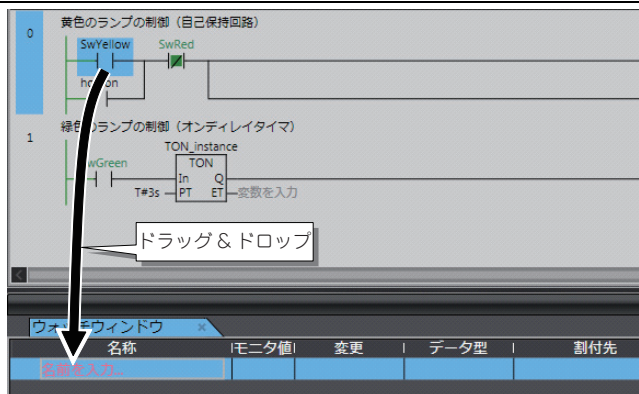
変数	説明	入力例
変数 LmpYellow	グローバル変数	LmpYellow
変数 holdon	Program0 のローカル変数	Program0.holdon
変数 Q	Program0 のローカル変数 TON_instance の出力変数	Program0.TON_instance.Q

方法2：ドラッグ＆ドロップ

ラダーエディタの変数 SwYellow の接点をクリックしてください。



[名前を入力...] へドラッグ＆ドロップしてください。



変数 SwYellow がウォッチウィンドウに登録されます。



参考

プログラム名（FB の場合はインスタンス名）を登録すると、登録したプログラムのすべてのローカル変数が階層構造で登録されます。

名称	モニタ値	変更	データ型	割付先	データ型式
▼ Program0			Program0		
holdon	False	TRUE FALSE	BOOL		Boolean ▼
LmpGreen	False	TRUE FALSE	BOOL		Boolean ▼
LmpYellow	False	TRUE FALSE	BOOL		Boolean ▼
P_CY	False	TRUE FALSE	BOOL		Boolean ▼
P_First_RunMode	False	TRUE FALSE	BOOL		Boolean ▼

強制値リフレッシュ

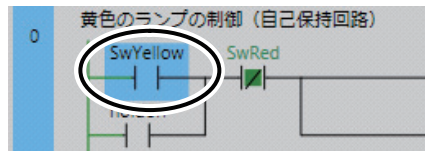
デジタルI/Oスレーブを使用するシステム構成でオンラインデバッグする場合は、強制値リフレッシュを使用して押しボタンスイッチからの入力値を変更し、デバッグします。

● 黄色の押しボタンスイッチの入力値を強制的に ON する

ラダーエディタで強制値リフレッシュをします。

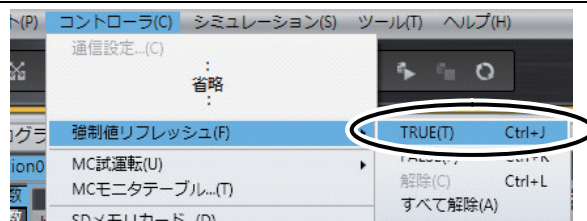
1 黄色の押しボタンスイッチのデバイス変数 SwYellow を選択します。

ラダーエディタの変数 SwYellow の接点をクリックしてください。



2 以下のいずれかの方法で、黄色の押しボタンスイッチからの入力を強制的に ON します。

方法 1：
メインメニューから [コントローラ] - [強制値リフレッシュ] - [TRUE] を選択してください。



方法 2：
右クリックメニューの [強制値リフレッシュ] - [TRUE] を選択してください。

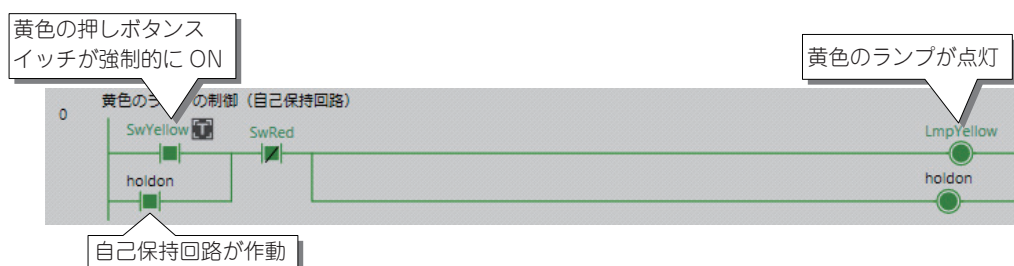


方法 3：

[Ctrl] キー + [J] キーを押してください。^{*1}

*1. 入力を強制的に OFF する場合は、[Ctrl] キーを押しながら [K] キーを押してください。

黄色の押しボタンスイッチからの入力値が強制的に ON となり、黄色のランプが点灯します。同時に自己保持回路が作動します。



● 黄色の押しボタンスイッチの入力値の強制状態を解除する

ラダーエディタで強制状態を解除します。

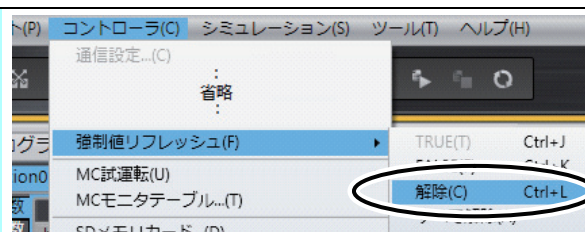
1 黄色の押しボタンスイッチのデバイス変数 SwYellow を選択します。

ラダーエディタの変数 SwYellow の接点をクリックしてください。

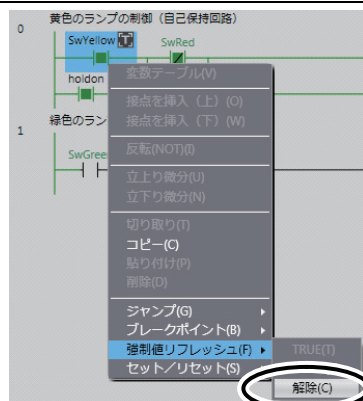


2 以下のいずれかの方法で、黄色の押しボタンスイッチからの入力値の強制状態を解除します。

方法 1：
メインメニューから [コントローラ] - [強制値リフレッシュ] - [解除] を選択してください。



方法 2：
右クリックメニューの [強制値リフレッシュ] - [解除] を選択してください。



方法 3：

 [Ctrl] キー +  [L] キーを押してください。

黄色の押しボタンスイッチの入力値が強制状態から解除されると、入力値は OFF となります。本ガイドの構成では、押しボタンスイッチは配線していないので、入力は常に OFF となります。自己保持回路が作動しているので、黄色のランプは点灯のままとなります。

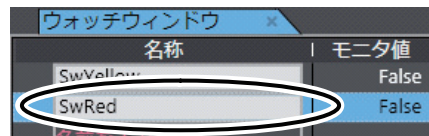


● 赤色の押しボタンスイッチの入力値を強制的に ON する

ウォッチウィンドウで強制値リフレッシュをします。あらかじめ、変数 SwRed をウォッチウィンドウに登録しておいてください。

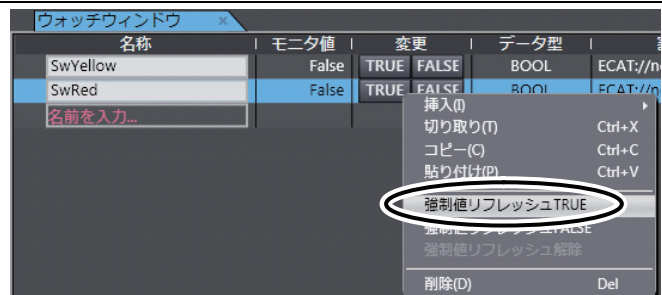
1 赤色の押しボタンスイッチのデバイス変数 SwRed を選択します。

ウォッチウィンドウの変数 SwRed をクリックしてください。



2 赤色の押しボタンスイッチからの入力を強制的に ON します。

右クリックメニューの「強制値リフレッシュ TRUE」を選択してください。



赤色の押しボタンスイッチからの入力値が強制的に ON になると、自己保持がリセットされ黄色のランプが消灯します。

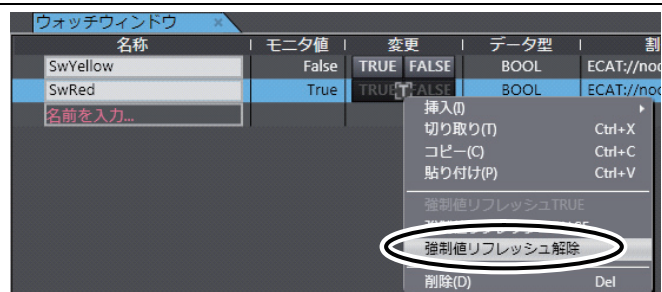


● 赤色の押しボタンスイッチの入力値の強制状態を解除する

ウォッチウィンドウで強制状態を解除します。

1 赤色の押しボタンスイッチからの入力値の強制状態を解除します。

右クリックメニューの「強制値リフレッシュ解除」を選択してください。





参考

- 上記のラダーエディタ、ウォッチウィンドウ以外に、I/O マップで強制値リフレッシュをする方法があります。I/O マップでは、変数の定義やアルゴリズムの作成をしていなくても強制値リフレッシュができます。このため、I/O マップでの強制値リフレッシュは、EtherCAT スレーブや CJ ユニットの入出力の配線を確認するときに便利です。操作方法については、□「A-6 I/O マップでの強制値リフレッシュ」(P.A-16)を参照してください。
- メインメニューの[コントローラ] - [強制値リフレッシュ] - [すべて解除]ですべての強制状態が解除できます。

セット／リセット

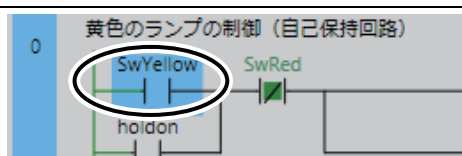
オフラインデバッグする場合、および、デジタル I/O スレーブを使用しないシステム構成でオンラインデバッグする場合は、セット／リセットを使用して押しボタンスイッチからの入力値を変更します。

● 黄色の押しボタンスイッチの入力値をセット／リセットする

ラダーエディタでセット／リセットする方法を説明します。

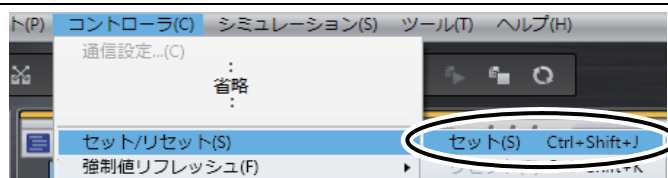
1 黄色の押しボタンスイッチのデバイス変数 SwYellow を選択します。

ラダーエディタの変数 SwYellow の接点をクリックしてください。

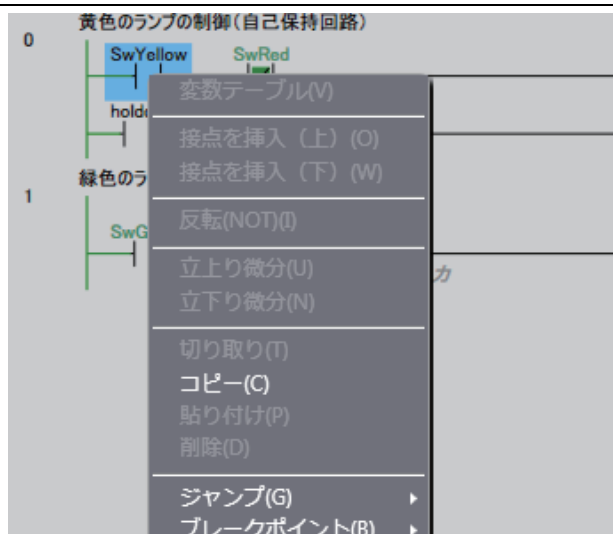


2 以下のいずれかの方法で、黄色の押しボタンスイッチからの入力をセット／リセットします。

方法 1：
メインメニューから [コントローラ] - [セット／リセット] - [セット] または [リセット] を選択してください。



方法 2：
右クリックメニューの [セット／リセット] - [セット] または [リセット] を選択してください。



方法3:
セット

Ctrl

[Ctrl] キー +

Shift

[Shift] キー +

J

[J] キーを押してください。

リセット、

Ctrl

[Ctrl] キー +

Shift

[Shift] キー +

K

[K] キーを押してください。

黄色の押しボタンスイッチからの入力をセットすると、変数 LmpYellow が ON します。その後黄色の押しボタン、スイッチからの入力をリセットしても、自己保持回路が作動しているので変数 LmpYellow は ON のままとなります。

● 赤色の押しボタンスイッチをセット／リセットする

ウォッチウィンドウでセット／リセットする方法を説明します。あらかじめ、変数 SwRed をウォッチウィンドウに登録しておいてください。

1 赤色の押しボタンスイッチのデバイス変数 SwRed を選択します。

ウォッチウィンドウの変数 SwRed をクリックしてください。

ウォッチウィンドウ

名称	モニタ値
SwYellow...	False
SwRed	False

2 セットする場合は、[変更] 列の [TRUE] を選択してください。リセットする場合は、[変更] 列の [FALSE] を選択してください。

• セット

TRUE をクリックしてください。

• リセット

FALSE をクリックしてください。

ウォッチウィンドウ

名称	モニタ値	変更
SwYellow	True	TRUE FALSE
SwRed	False	TRUE FALSE

自己保持回路が作動している状態で、赤色の押しボタンスイッチからの入力をセットすると、自己保持がリセットされて変数 LmpYellow が OFF します。

4-6-4 データトレースでランプの動作を確認する

データトレース機能を使用して、緑色のランプの動作（オンディレイタイマ）を確認します。オンラインデバッグでもオフラインデバッグでも操作方法は同じです。



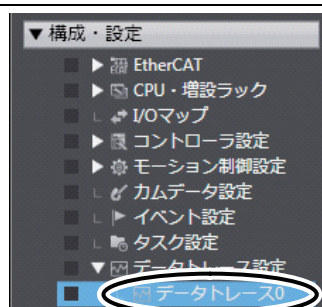
データトレースを設定する

データトレースを起動し、トレースタイプとトリガ条件、サンプリング変数を設定します。

● データトレースを起動する

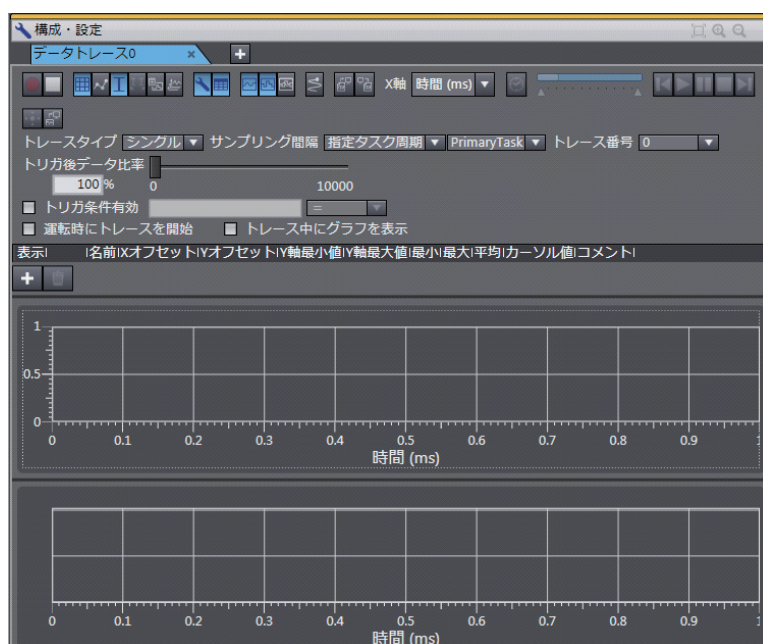
1 データトレース設定を作成します。

マルチビューエクスプローラで「構成・設定」－「データトレース設定」を右クリックし、「追加」－「データトレース」を選択してください。



マルチビューエクスプローラに「データトレース 0」が作成されます。

2 「データトレース 0」をダブルクリックしてください。 データトレース画面が表示されます。



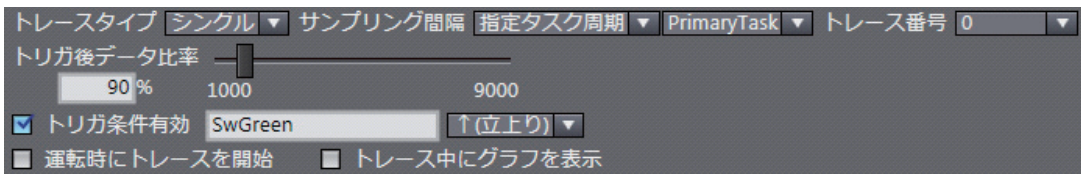
データトレース操作、画面表示の ON/OFF などを指示するツールバーです。

データトレースのトリガ条件やサンプリングする変数を設定します。

データトレースの実行結果が表示されます。

● トレースタイプとトリガ条件を設定する

トレースタイプとトリガ条件を設定します。トレースタイプとトリガ条件は以下のとおりです。



設定項目	設定値	説 明
トレースタイプ	シングル	トリガ条件成立前後のデータをサンプリングします。サンプリング数は、1 変数あたり 10000 点です。ここでは、トリガ条件成立前後のデータをサンプリングするため、トレースタイプは「シングル」を使用します。
サンプリング間隔	指定タスク周期、PrimaryTask	プライマリ定周期タスクのタスク周期に 1 回サンプリングを実行します。ここでは、プライマリ定周期タスクのタスク周期を 1ms に設定していますので、10 秒分のデータがサンプリングできます。(1ms×10000 点)
トリガ後データ比率	90%	トリガ成立前のサンプリングデータを 10%、トリガ成立後のサンプリングデータを 90% の割合とします。
トリガ条件	有効、変数 SwGreen の立上り	変数 SwGreen の立上りをトリガ条件とします。

トレースタイプとトリガ条件の設定手順を説明します。

1 トレースタイプ選択コンボボックスで「シングル」を選択してください。

コンボボックスをクリックして、「シングル」を選択してください。

トレースタイプ

2 サンプリング間隔コンボボックスで「指定タスク周期」、「PrimaryTask」を選択してください。

コンボボックスをクリックして、「指定タスク周期」、「PrimaryTask」を選択してください。

サンプリング間隔

3 以下のいずれかの方法で、トリガ後データ比率を「90%」に設定します。

方法 1：
トリガ後データ量スライダをドラッグし、90% までスライドしてください。

トリガ後データ比率

方法 2：
トリガ後データ量入力エディットボックスに、「90」を入力してください。

トリガ後データ比率

4 トリガ条件を有効にして、「変数 SwGreen」の立上りをトリガ条件として設定します。

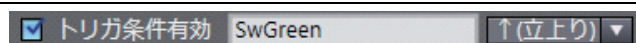
トリガ条件有効チェックボックスをチェックしてください。



エディットボックスに、「SwGreen」と入力



コンボボックスをクリックして、「↑（立上り）」を選択



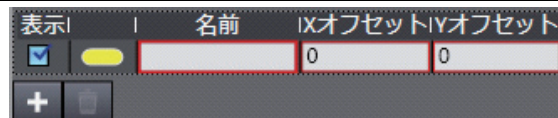
● トレース対象変数を設定する

データトレース対象の変数を設定します。

ここでは、変数 SwGreen と変数 LmpGreen をトレース対象変数として設定します。

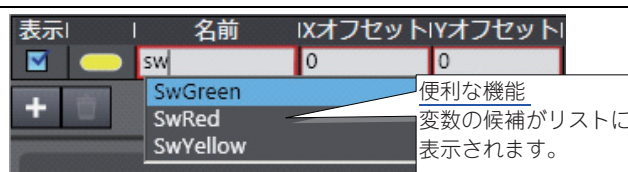
1 新規トレース対象変数行を追加選択します。

をクリックしてください。



2 変数 SwGreen を追加します。

名前列をクリックして、「swgreen」と入力してください。



変数 SwGreen が登録されます。



3 同様に、変数 LmpGreen を追加してください。

表示		名前	IXオフセット	IYオフセット
☒		SwGreen	0	
☒		LmpGreen	0	

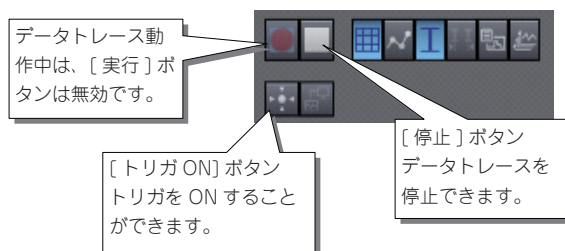
データトレースする

データトレースを開始し、結果を確認します。

あらかじめ、変数 SwGreen と変数 LmpGreen をウォッチウィンドウに登録しておいてください。

● データトレースの開始

1 [実行] ボタンをクリックしてください。



2 緑色の押しボタンスイッチのデバイス変数 SwGreen をウォッチウィンドウで ON してください。

デジタル I/O スレーブを使用するシステム構成でオンラインデバッグする場合は、「強制値リフレッシュ」で ON にしてください。オフラインデバッグする場合、および、デジタル I/O スレーブを使用しないシステム構成でオンラインデバッグする場合は、「セット/リセット」で ON にしてください。

トリガが成立し、サンプリング数が 10000 に達すると、データトレース結果が表示されます。

● データトレース結果の表示

以下のようにデータトレース画面に結果が表示されます。



参考

データトレース画面は、ツールボタンで必要な情報だけ表示することができます。
上記の画面では、以下の設定としています。

非表示：グリッド表示、カーソル表示、トレース設定、アナログチャート
表示：凡例

4-6-5 オンラインエディットでロジックを変更する

オンラインエディットとは、CPU ユニット内のユーザプログラムの一部を、Sysmac Studio から直接的に変更、あるいは追加する機能です。

以下を選択してオンラインエディットすることができます。

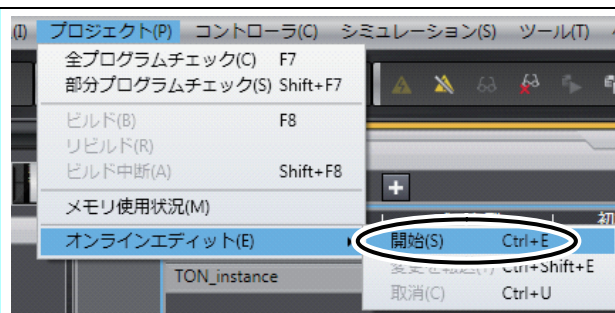
- ラダーの選択セクション
- ST 言語で記述された POU（プログラム、ファンクションブロック、ファンクション）
- グローバル変数

ここでは、緑色のランプの制御に使用しているオンディレイタイマ命令の設定時間を、3 秒から 5 秒に変更します。オンラインデバッグでもオフラインデバッグでも操作方法は同じです。

1 Program0 の Section0 を開いてください。

2 以下のいずれかの方法で、オンラインエディットを開始します。

方法 1：
メインメニューから [プロジェクト] -
[オンラインエディット] - [開始] を選択
してください。



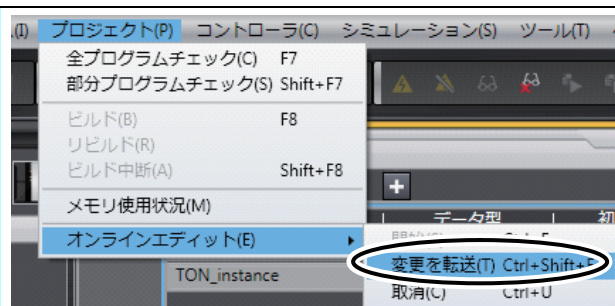
方法 2：

 [Ctrl] キー +  [E] キーを押
してください。

3 ラダーエディタ上で、オンディレイタイマ命令の入力変数 PT のパラメータを「T#5s」に変更
します。

4 以下のいずれかの方法で、変更内容を転送します。

方法 1：
メインメニューから [プロジェクト] -
[オンラインエディット] - [変更を転送]
を選択してください。

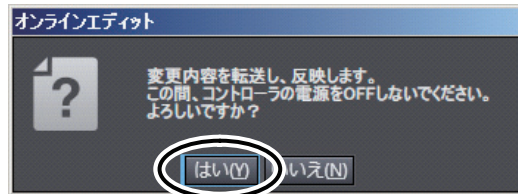


方法2:

 [Ctrl] キー +  [Shift] キー
+  [E] キーを押してください。

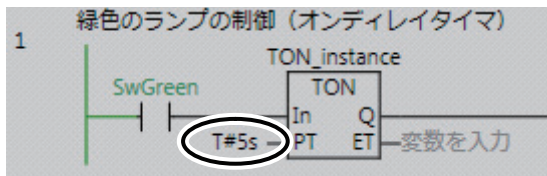
5 反映確認のダイアログが表示されますので、「はい」をクリックしてください。

 をクリックしてください。



CPU ユニット内のユーザプログラムのロジックが変更されます。

これで、オンラインエディットが完了しました。



5

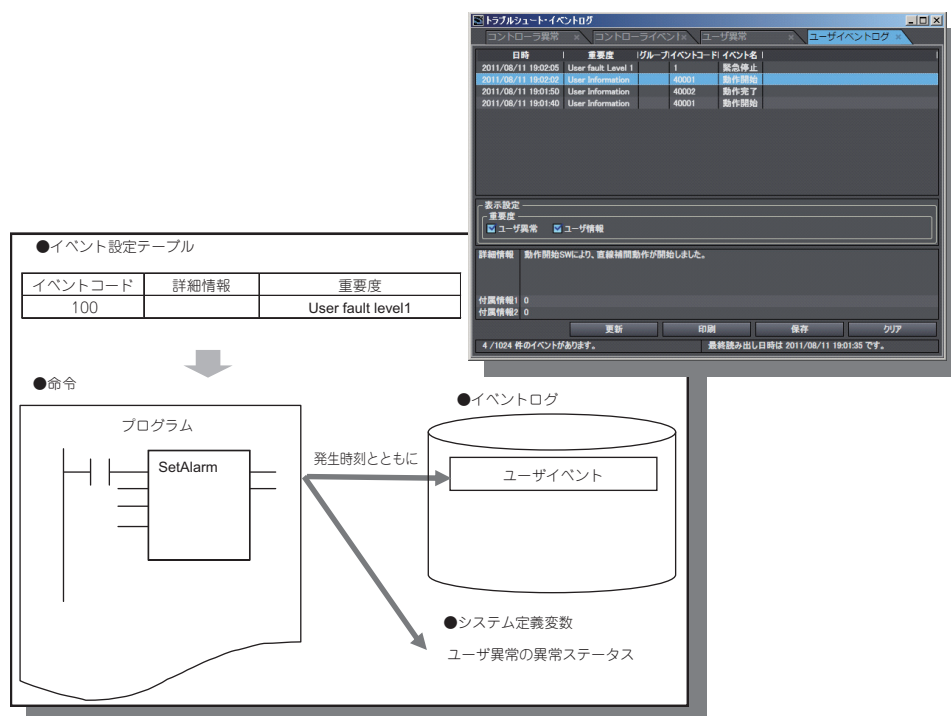
便利な機能

本章では、NJ シリーズ CPU ユニットおよび Sysmac Studio を使用するとき便利な機能を紹介します。

5-1	アプリケーションのイベントを登録し、管理する	5-2
5-2	ユーザプログラム資産を守る	5-5
5-2-1	「ユーザプログラム実行用 ID 認証」による盗用防止	5-5
5-2-2	「ユーザプログラム復元情報なし転送」による盗用防止	5-6
5-2-3	「プロジェクトファイル全体のプロテクト機能」によるユーザ資産情報の保護	5-6
5-2-4	「データプロテクト機能」によるユーザ資産情報の保護	5-6
5-2-5	「操作権限認証」による誤操作の防止	5-7
5-2-6	「書込プロテクト機能」による Sysmac Studio からの書き込み操作禁止	5-8
5-2-7	「CPU ユニット名称機能」による Sysmac Studio の誤接続防止	5-8

5-1 アプリケーションのイベントを登録し、管理する

「ユーザイベント機能」を使用すると、ユーザが設計するアプリケーション上で発生するさまざまな事象を、「ユーザイベント」として登録、管理できます。「ユーザイベント」は、Sysmac Studio のイベント設定テーブルに登録し、ユーザイベント命令を使って発生させます。発生した「ユーザイベント」は、Sysmac Studio、または、NJ シリーズ対応 NS シリーズ表示器で確認できます。



ユーザイベント機能の特長

ユーザイベント機能には、以下の特長があります。

- 用途に合わせて、イベント重要度（８段階のユーザ異常と、ユーザ情報）を使い分けできます。
- イベントの発生箇所や分類などを表すグループ名を指定できます。
- NJ シリーズで定義している異常／情報であるコントローライイベントと合わせて、記録と管理ができます。
- Sysmac Studio、または、NJ シリーズ対応 NS シリーズ表示器で、時系列で確認できます。

ユーザイベントログの使い方

ユーザイベントを設定する手順と確認方法を以下に説明します。

● ユーザイベントを設定する

1 マルチビューエクスプローラの[構成・設定]－[イベント設定]をダブルクリックしてください。

2 イベント設定テーブルにユーザイベントを登録してください。

イベントコード	イベント名	重要度	グループ	詳細情報
1	緊急停止	User fault Level 1		緊急停止SWにより、直線補間動作が緊急停止しました。
2	40001 動作開始	User Information		動作開始SWにより、直線補間動作が開始しました。
3	40002 動作完了	User Information		直線補間動作が完了しました。



参考

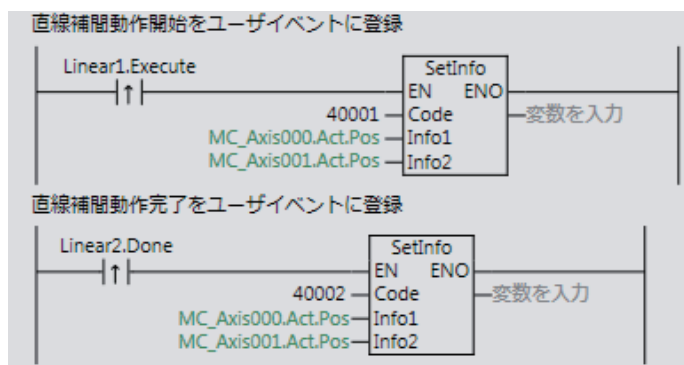
イベント設定テーブルの登録内容は、Microsoft Excel で編集することができます。詳細は、□『Sysmac Studio Version 1 オペレーションマニュアル (SBCA-470)』を参照してください。

	A	B	C	D	E	F
1	イベントコード	イベント名	重要度	グループ	詳細情報	
2	40001	動作開始	User Information		動作開始SWにより、直線補間動作が開始しました。	
3	40002	動作完了	User Information		直線補間動作が完了しました。	
4	1	緊急停止	User fault Level 1		緊急停止SWにより、直線補間動作が緊急停止しました。	
5						切り取り(I)
6						コピー(C)
7						貼り付け(P)

3 ユーザイベント命令でプログラミングしてください。

ユーザ異常：ユーザ異常発生 (SerAlarm) 命令を使用します。

ユーザ情報：ユーザ情報発生 (SetInfo) 命令を使用します。



使用上の注意

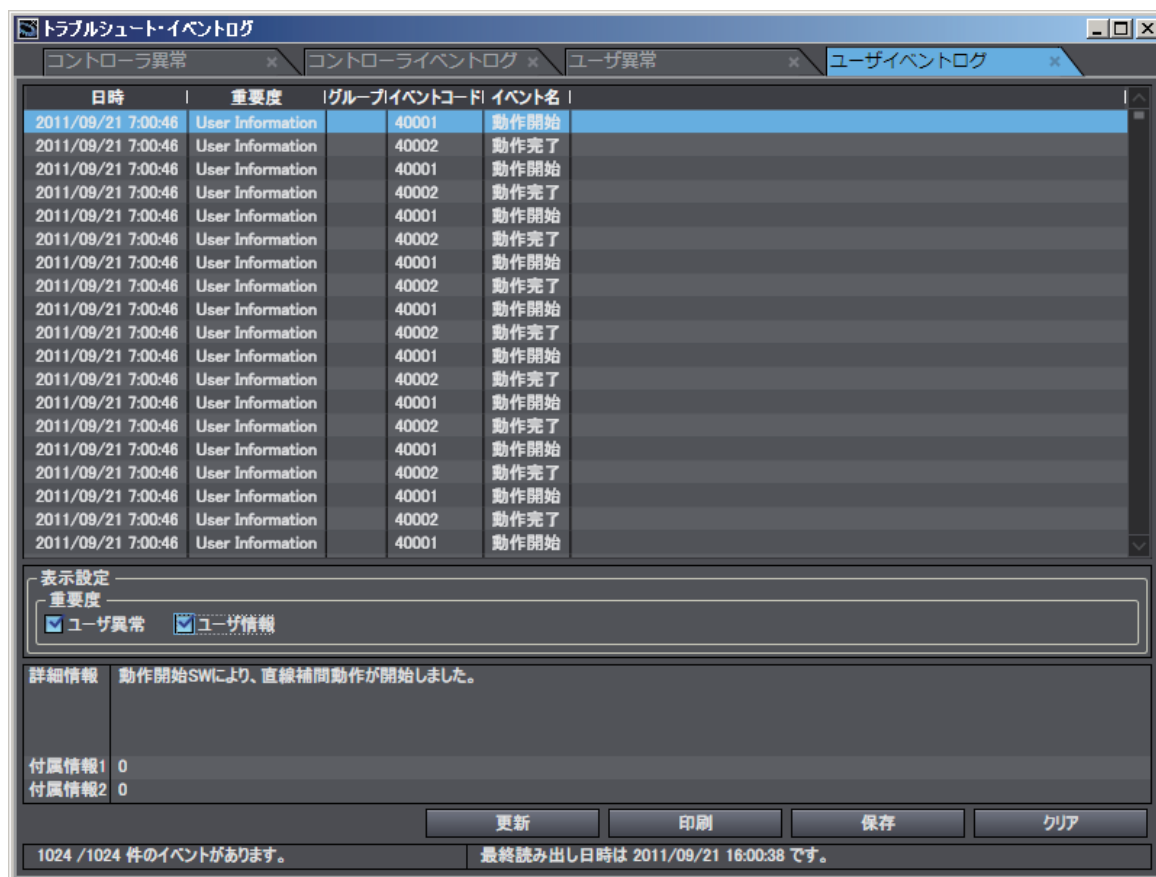
ユーザ異常発生 (SerAlarm) 命令、ユーザ情報発生 (SetInfo) 命令に使用する「Info1」(付属情報 1)、「Info2」(付属情報 2) には、必ず変数を指定してください。定数を指定すると、ビルド時に異常になります。付属情報を使用しない場合は、ダミーの変数を指定してください。

4 イベント設定テーブルとユーザプログラムを CPU ユニットに転送してください。

● ユーザイベントの発生を確認する

1 オンライン接続状態で、[ツール] - [トラブルシュート・イベントログ] を選択してください。または、ツールバーの [トラブルシュート、異常履歴] ボタンをクリックしてください。

[ユーザ異常] タブおよび [ユーザイベントログ] タブでユーザイベントの発生状況が確認できます。



5-2 ユーザプログラム資産を守る

セキュリティ機能を使って、ユーザプログラム資産を守ることができます。セキュリティ機能には、以下の機能があります。ここでは、セキュリティ機能の概要を説明します。

セキュリティ機能	目的
ユーザプログラム実行用 ID 認証	資産の盗用防止
ユーザプログラム復元情報なし転送	
プロジェクトファイル全体のプロテクト機能	
データプロテクト機能 ^{*1}	
操作権限認証	誤操作防止
書き込みプロテクト機能	
CPU ユニット名称機能	誤接続防止

*1. CPU ユニット Ver.1.01 以降と Sysmac Studio Ver.1.02 以降の組み合わせで使用可能です。



参考

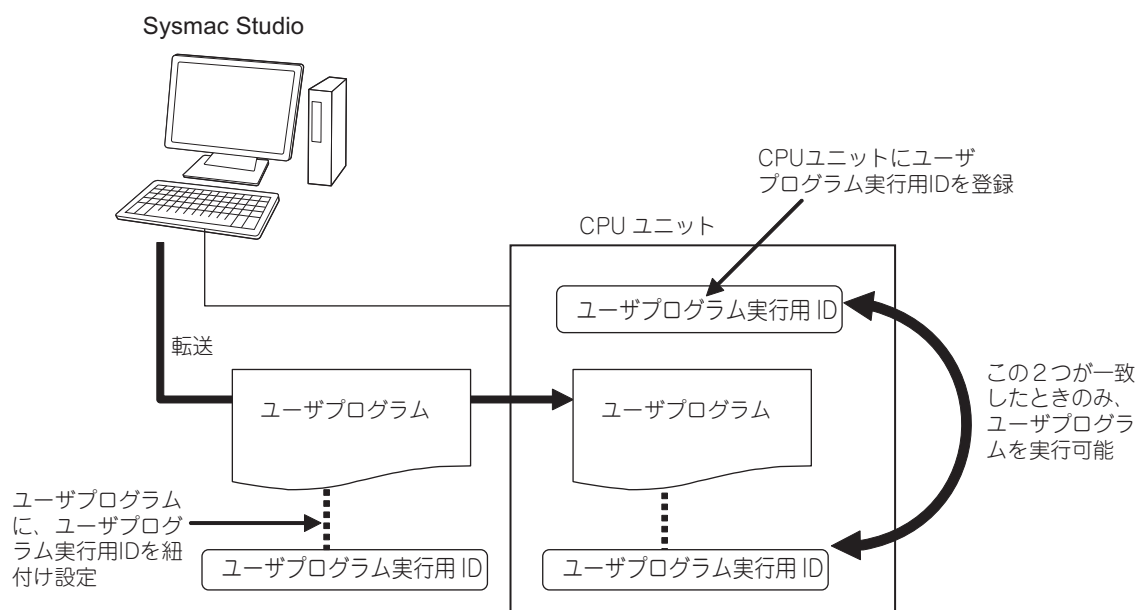
セキュリティ機能の詳細については、□『Sysmac Studio Version 1 オペレーションマニュアル(SBCA-470)』、□『NJ/NX シリーズ CPU ユニット ユーザーズマニュアル ソフトウェア編 (SBCA-467)』を参照してください。

5-2-1 「ユーザプログラム実行用 ID 認証」による盗用防止

あらかじめコントローラに特定の ID を入力しておく、同じ ID を紐付け設定したユーザプログラムだけを実行可能にできます。この ID を「ユーザプログラム実行用 ID」と呼びます。

「ユーザプログラム実行用 ID 認証」により、以下のような使用制限ができます。

- あるコントローラでは、特定のユーザプログラムしか実行できない。
- あるユーザプログラムを、異なるコントローラでは使用できない。





使用上の注意

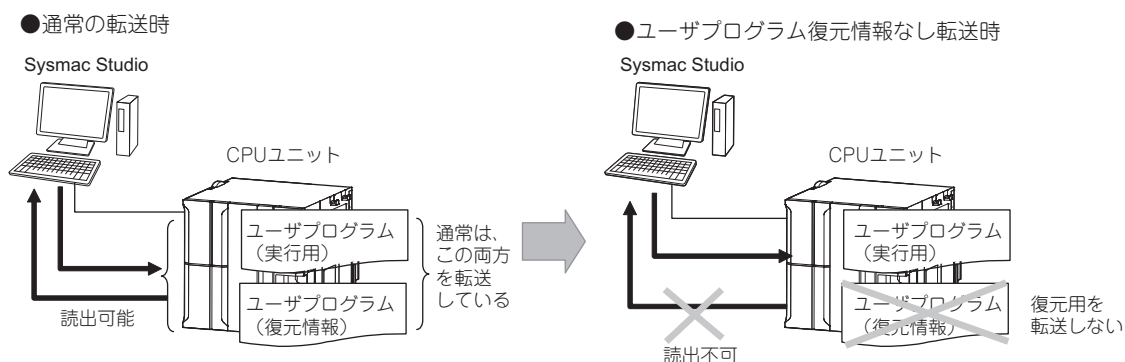
- ユーザプログラムへのユーザプログラム実行用 ID の設定は、1 度しかできません。
- ユーザプログラム実行用 ID を忘れないよう、メモしておいてください。
- ユーザプログラム実行用 ID を設定する前に、未設定状態のプロジェクトファイルをバックアップしておくことをお勧めします。

5-2-2 「ユーザプログラム復元情報なし転送」による盗用防止

Sysmac Studio は、CPU ユニットにユーザプログラムを転送するときは、CPU ユニットからユーザプログラムを復元するための情報も転送しています。

「ユーザプログラム復元情報なし転送」を使うと、ユーザプログラムを転送するときに復元するための情報を転送しないことで、ユーザプログラムを読み出すことを禁止できます。

現場でユーザプログラムを保守することがなく、かつユーザプログラムデータの盗用を防止したいときに使用します。



5-2-3 「プロジェクトファイル全体のプロテクト機能」によるユーザ資産情報の保護

プロジェクトファイルをエクスポートするときに、プロジェクトファイルにパスワードをかけることができます。これにより、プロジェクトファイルを暗号化し、ユーザ資産を保護します。

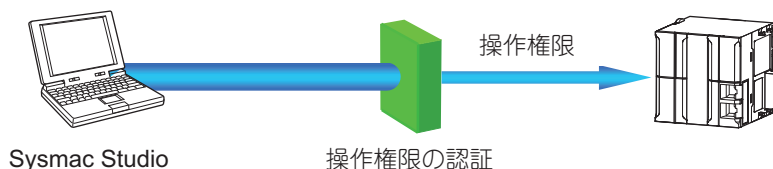
5-2-4 「データプロテクト機能」によるユーザ資産情報の保護

プロジェクトのデータのうち保護したいデータにプロテクトをかけることにより、ユーザ資産を保護します。

5-2-5 「操作権限認証」による誤操作の防止

操作ミスによって装置や人に危害を加える可能性がある場合に、操作権限に応じてオンラインの操作を制限する機能です。Sysmac Studio で CPU ユニットに「操作権限認証」のパスワードをあらかじめ登録しておきます。オンライン接続時にパスワードを入力することで、操作権限カテゴリに応じた操作が可能になります。

管理者が操作権限ごとにパスワードを設定し、スキルに応じて操作権限名とパスワードを使用者に連絡することで運用します。



● 操作権限の種類

Sysmac Studio の操作権限の種類は、権限の大きい順に以下の 5 種類が可能です。

日本語名	英語名	パスワード設定
管理者	Administrator	必要
設計者 ^{*1}	Planning Engineer	任意 ^{*2} [操作権限認証設定] ダイアログボックスのパスワード入力を省略時の操作権限で要否を設定します。
保全者	Maintenance	
操作者 ^{*1}	Operator	
観測者 ^{*1}	Observer	不要

*1. CPU ユニット Ver.1.01 以降と Sysmac Studio Ver.1.02 以降の組み合わせで使用可能です。

*2. [操作権限認証設定] ダイアログボックスのパスワード入力を省略時の操作権限で要否を設定します。省略時の操作権限よりも上位の権限にはパスワードが必要です。同じまたは下位の権限にはパスワードが不要です。

● オンライン操作の操作権限の一例

(○ : 操作可能、△ : 操作時に都度認証実施、× : 操作不可能)

ステータスモニタ機能 (例)	管理者	設計者	保全者	操作者	観測者
異常モニタ (トラブルシュート)	○	○	○	○	○
I/O モニタ機能 (例)	管理者	設計者	保全者	操作者	観測者
I/O モニタ (読出)	○	○	○	○	×
I/O モニタ (書込)	○	○	○	△	×
セット/リセット	○	○	○	△	×
強制値リフレッシュ	○	○	○	×	×
コントローラ操作機能 (例)	管理者	設計者	保全者	操作者	観測者
運転モード/プログラムモード	○	○	△	×	×
オンラインエディット	○	○	△	×	×
コントローラリセット	○	○	×	×	×
異常解除 (トラブルシュート)	○	○	○	△	×
軸試運転、軸試運転再開	○	○	△	×	×
ユーザプログラム実行用 ID (コントローラ)	○	×	×	×	×
CPU ユニットの書込プロテクト	○	○	○	×	×

● パスワード仕様

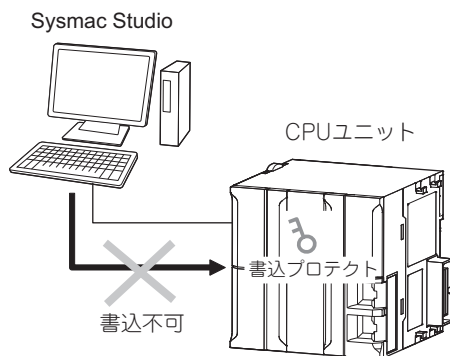
項目	内容
有効文字数	8 文字以上かつ 32 文字以下
使用可能文字	半角英数字 (大文字、小文字は区別する)

5-2-6 「書込プロテクト機能」による Sysmac Studio からの書き込み操作禁止

Sysmac Studio からの書き込み操作を防止する機能です。

以下の2通りの方法で書き込みプロテクトを設定できます。

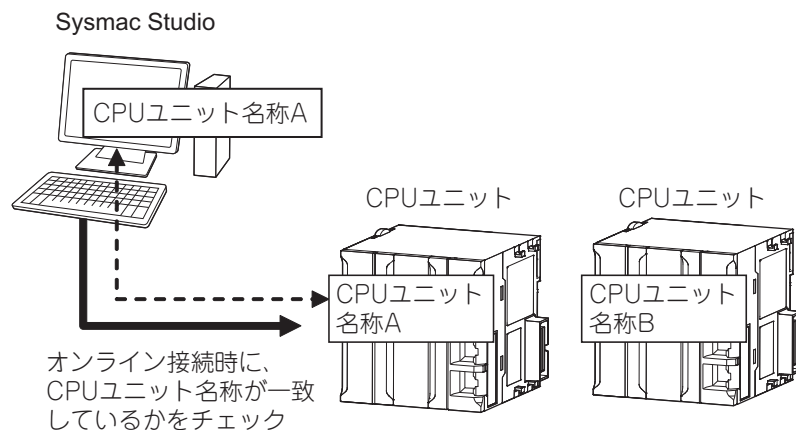
- CPU ユニットの電源投入時に、自動的に書込プロテクトを有効にする。
- Sysmac Studio からオンライン接続して、[コントローラ] メニューから [セキュリティ] – [書込みプロテクトスイッチ] にて、「セット」 / 「解除」を切り替える。

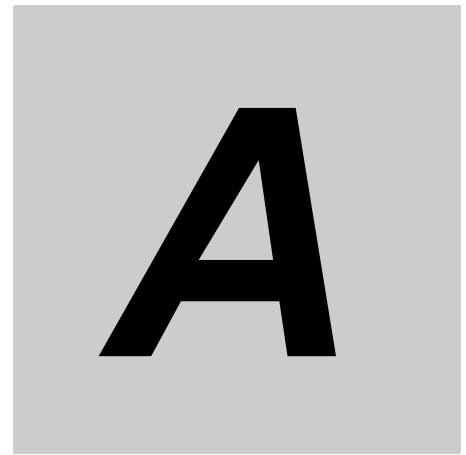


5-2-7 「CPU ユニット名称機能」による Sysmac Studio の誤接続防止

Sysmac Studio からのオンライン接続時に、プロジェクト上の CPU ユニット名称と接続先 CPU ユニットの CPU ユニット名称が一致しているかどうかを確認します。

これにより、Sysmac Studio から誤った CPU ユニットへの接続を防止することができます。とくに、EtherNet/IP ネットワークを介した接続のときに有効です。





付録

プログラミングからデバッグでよく使う操作などを説明します。

A-1	クロスリファレンスの操作方法	A-2
A-2	変数テーブル編集の便利な機能	A-4
A-3	よく使うプログラミング操作	A-6
A-4	オンラインのネットワーク構成の作成	A-13
A-5	オンラインデバッグとオフラインデバッグの違い	A-15
A-6	I/O マップでの強制値リフレッシュ	A-16

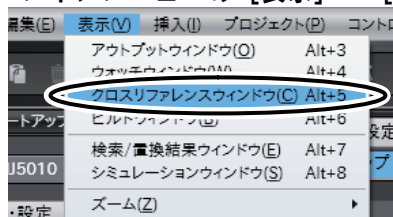
A

A-1 クロスリファレンスの操作方法

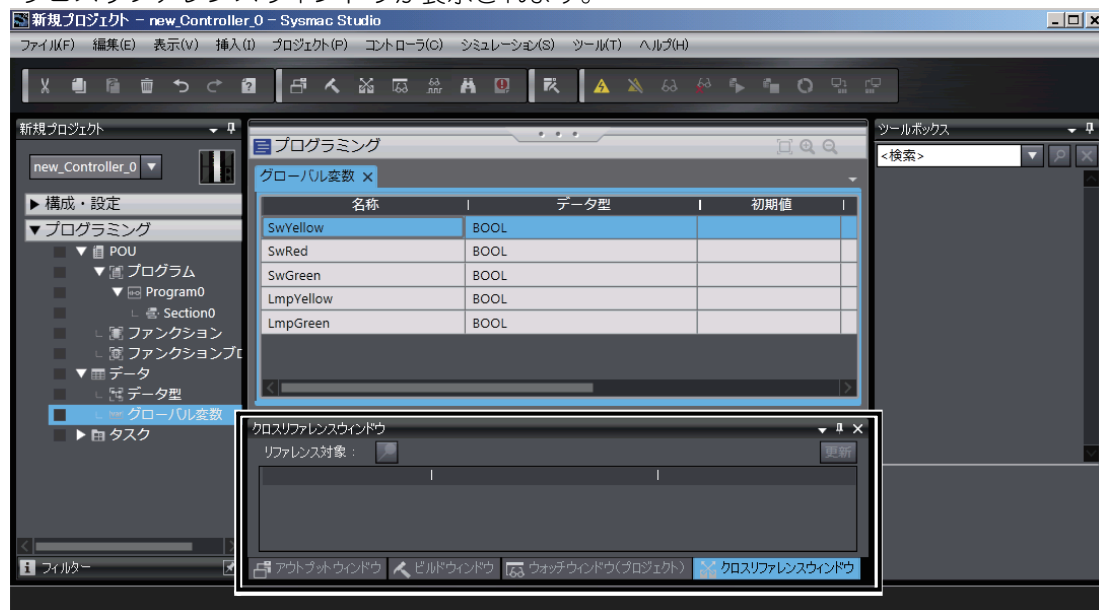
クロスリファレンスとは、変数や EtherCAT スレーブの I/O ポートなどが、プロジェクトのどこで使用されているかを一覧で表示する機能です。

ここでは、立ち上げ手順で作成したプロジェクトを使用して操作方法を説明します。
グローバル変数テーブルに登録されている変数 SwRed が、ラダーエディタのどこで使用されているかをクロスリファレンスで確認します。

- 1 グローバル変数テーブルを開いてください。
- 2 メインメニューの [表示] – [クロスリファレンスウィンドウ] を選択してください。

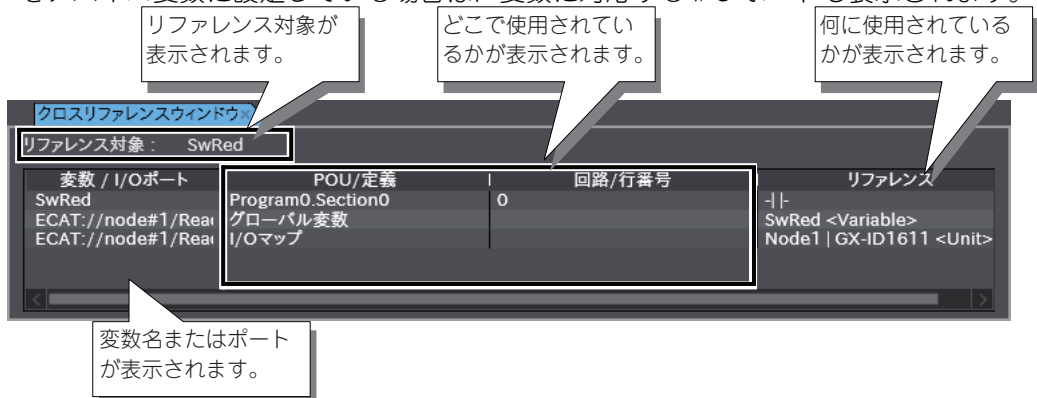


クロスリファレンスウィンドウが表示されます。

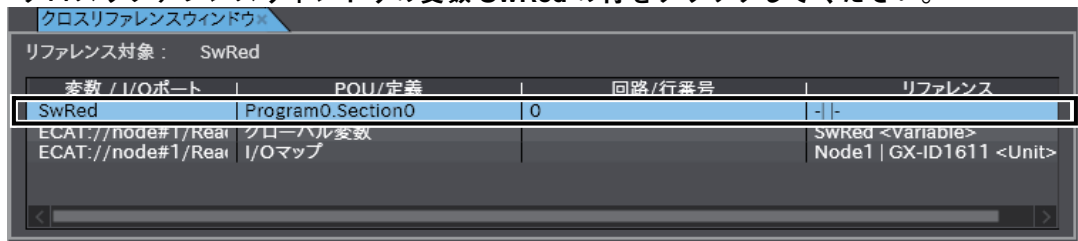


3 グローバル変数テーブルの変数 SwRed をクリックしてください。

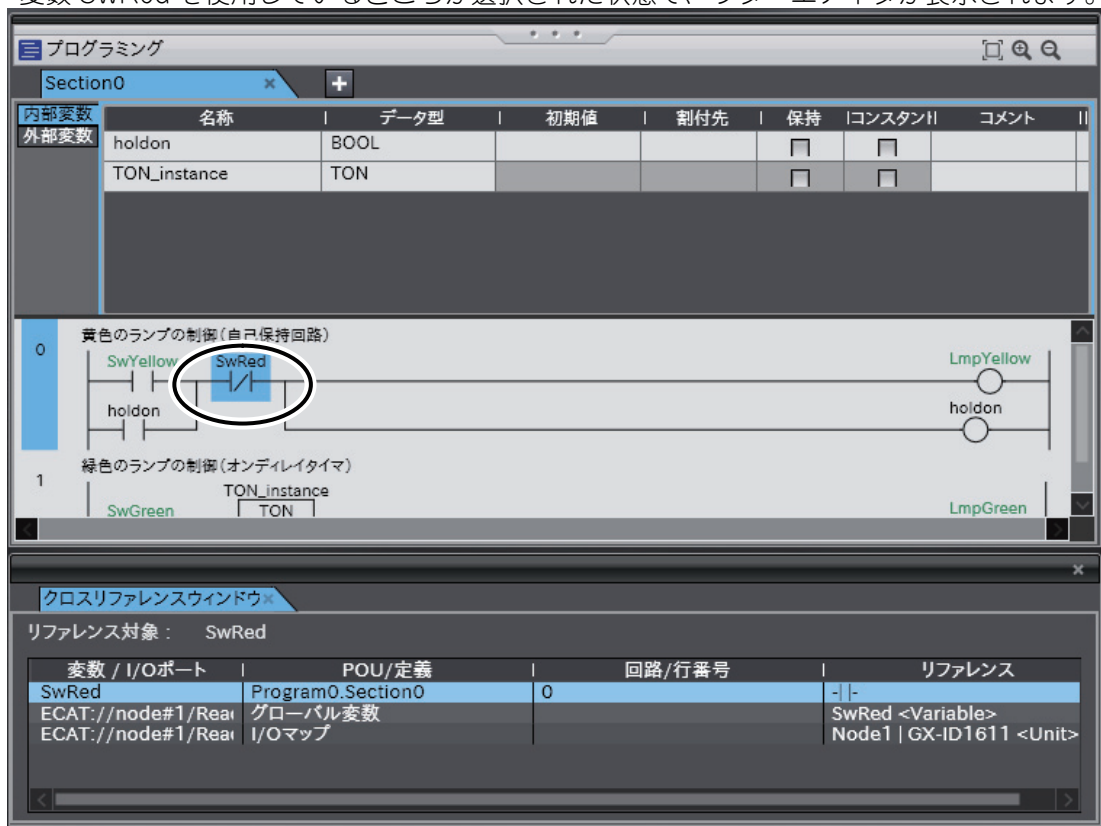
変数 SwRed を使用しているところが、クロスリファレンスウィンドウに表示されます。変数をデバイス変数に設定している場合は、変数に対応する I/O ポートも表示されます。



4 クロスリファレンスウィンドウの変数 SwRed の行をクリックしてください。



変数 SwRed を使用しているところが選択された状態で、ラダーエディタが表示されます。



参考

グローバル変数テーブル以外にも、ラダーエディタや I/O マップの変数をクリックすることで、クロスリファレンスのリファレンス対象にすることができます。

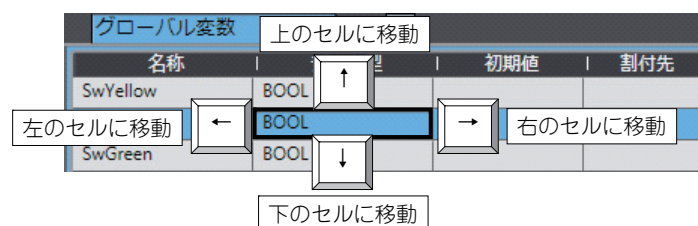
A-2 変数テーブル編集の便利な機能

変数テーブルの編集に便利な機能に、「直感的なキー操作による編集」、「名称入力アシスト機能」、「データ型入力アシスト機能」があります。

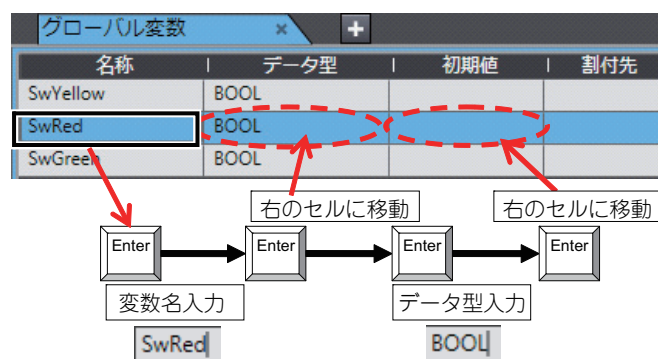
直感的なキー操作

直感的なキー操作により、変数の新規作成や編集をスムーズに行うことができます。

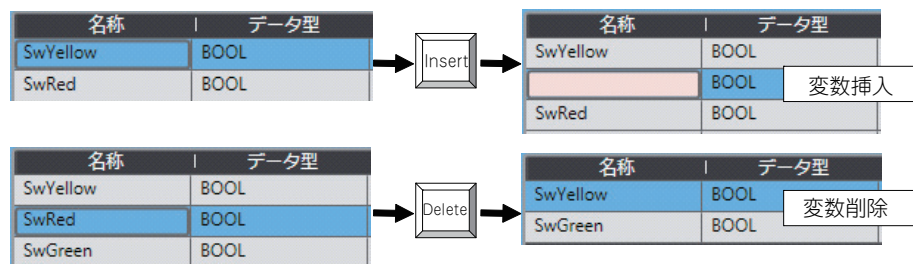
- [矢印] キーで任意のセルに移動



- [Enter] キーで右のセルに移動

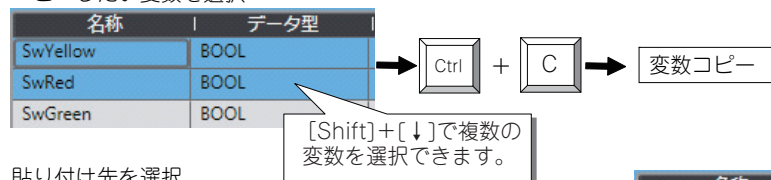


- [Insert] キーで変数挿入、[Delete] キーで変数削除

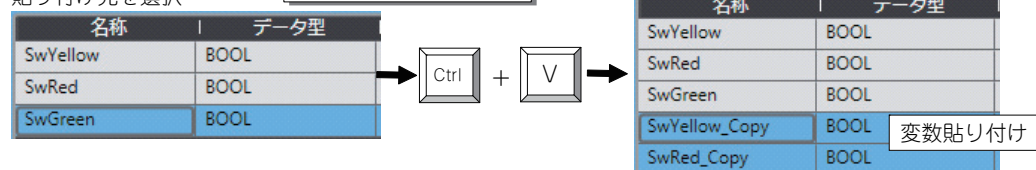


- [Ctrl] + [C] で変数コピー、[Ctrl] + [V] で変数貼り付け

コピーしたい変数を選択

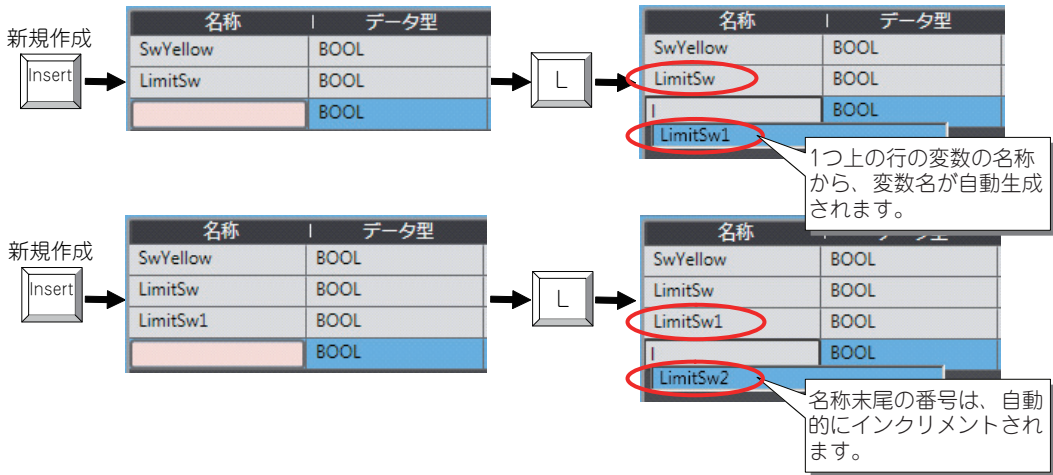


貼り付け先を選択



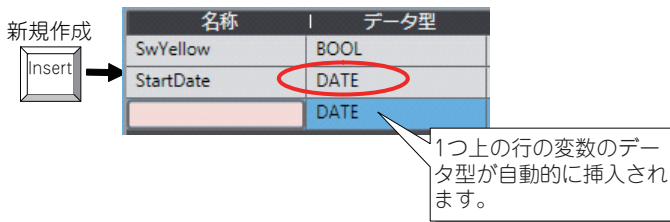
● 名称入力アシスト機能

変数を新規作成するときに、変数名称の末尾に番号が自動的に付加されます。
類似の変数を複数作成する場合に便利な機能です。

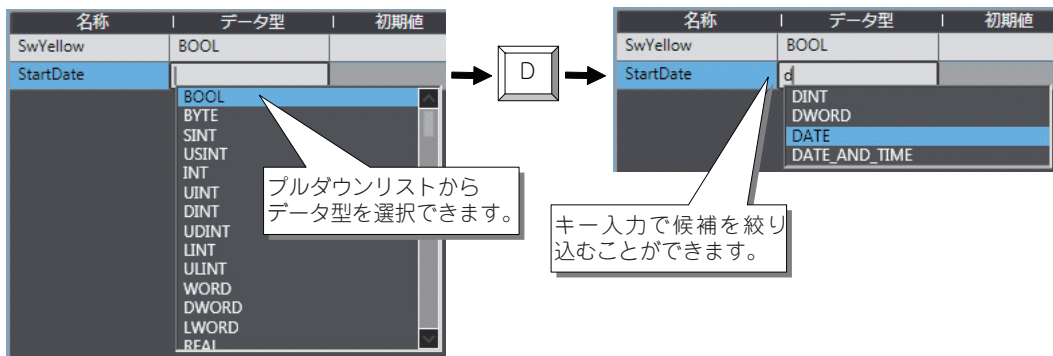


● データ型入力アシスト機能

- 変数を新規作成するときに、1つ上の行の変数のデータ型と同じデータ型が自動的に挿入されます。



- プルダウンリストからデータ型を選択することができます。



A-3 よく使うプログラミング操作

プログラミングでよく使う操作を紹介します。

微分型接点の入力

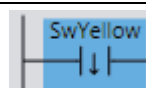
● 立ち上がり微分

接点をクリックして、右クリックメニューの「立ち上がり微分」を選択してください。



● 立ち下がり微分

接点をクリックして、右クリックメニューの「立ち下り微分」を選択してください。



FUN 型命令の立ち上がり微分オプションの入力

命令名の先頭に「@」を入力します。

接続線をクリックして、



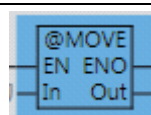
「I (アイ)」キーを押してください。



FUN が挿入され、ファンクション名が入力できるようになります。



「@MOVE」と入力してください。

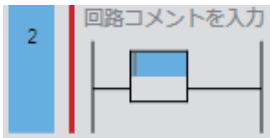


1 秒クロックパルスの入力

接続線をクリックして、



[I (アイ)] キーを押してください。



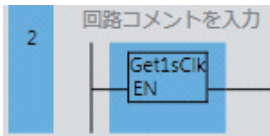
FUN が挿入され、ファンクション名が入力できるようになります。



「Get1sClk」と入力して、



[Enter] キーを 2 回押してください。



コイルを追加してください。



クロックパルスの周期は、100 μ s、1ms、10ms、20ms、100ms、1s、1min のいずれかです。

常時 ON の入力

接続線をクリックして、



[C] キーを押してください。



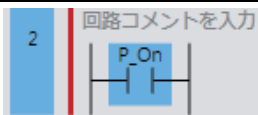
接点が挿入されます。



「P_On」と入力して、



[Enter] キーを 2 回押して確定してください。



運転開始時 1 周期 ON フラグの入力

接続線をクリックして、



[C] キーを押してください。



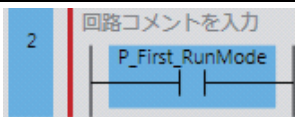
接点が挿入されます。



「P_First_RunMode」と入力して、

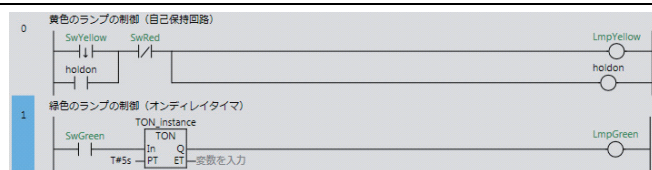


[Enter] キーを 2 回押して確定してください。



回路のコピー & ペースト

コピーする回路をクリックして選択してください。^{*1}



右クリックメニューの「コピー」を選択してください。

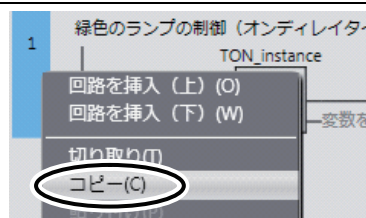
または、



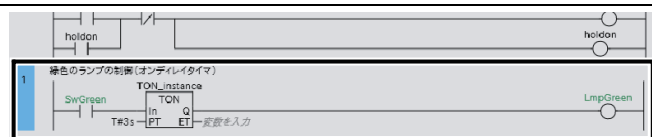
[Ctrl] キー +



[C] キーを押してください。



コピーしたい場所の回路をクリックして選択してください。



右クリックメニューの「貼り付け」を選択してください。

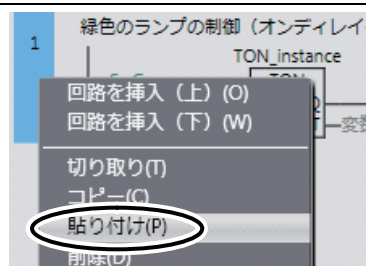
または、



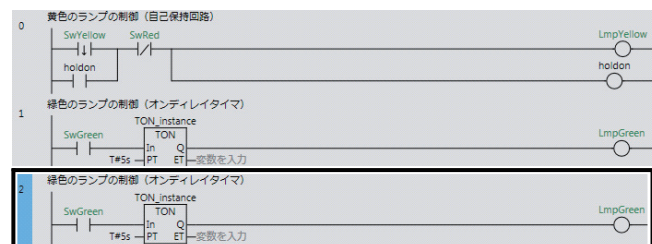
[Ctrl] キー +



[V] キーを押してください。



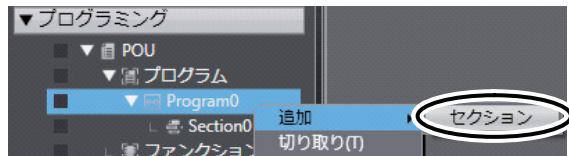
選択した回路の下に、コピーした回路が貼り付きます。



*1. 複数回路を選択する場合は、[Shift] キーを押しながら複数回路をクリックをしてください。

セクションの追加

マルチビューエクスプローラでセクションを追加するプログラムを選択し、右クリックメニューの「追加」－「セクション」を選択してください。



セクションが追加されます。



セクションのコピー & ペースト

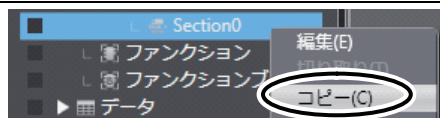
マルチビューエクスプローラでコピーするセクションを選択してください。



右クリックメニューの「コピー」を選択してください。

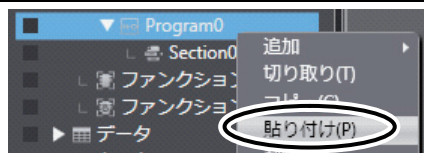
または、

[Ctrl] キー + [C] キーを押してください。

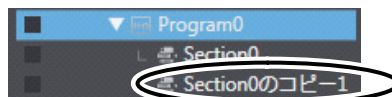


マルチビューエクスプローラでコピー先のプログラムを選択して、右クリックメニューの「貼り付け」を選択してください。

または、
 [Ctrl] キー + [V] キーを押してください。



選択したプログラムの一番下にセクションが貼り付きます。



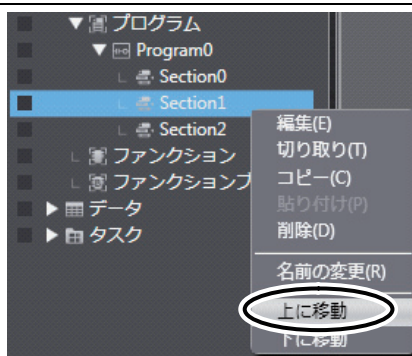
セクションの順番の変更

セクションは、マルチビューエクスプローラに表示される順にしたがって、上から下へ実行されます。実行順を変更するにはセクションの順序を変更します。
上に移動する場合を説明します。

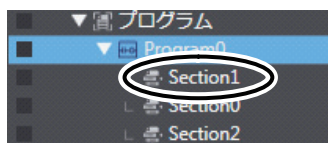
マルチビューエクスプローラで移動するセクションを選択してください。



右クリックメニューの「上に移動」を選択してください。
または、移動先にドラッグ＆ドロップしてください。



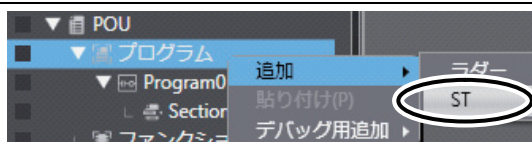
セクションが上に移動します。



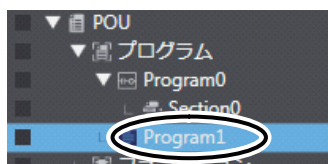
POU の追加

ST 言語のプログラムを追加する場合を説明します。

マルチビューエクスプローラでプログラムを選択し、右クリックメニューの「追加」→「ST」を選択



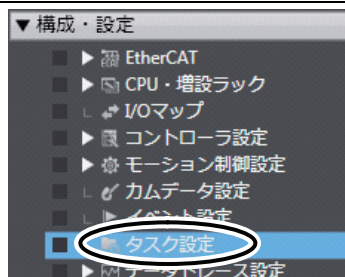
ST 言語のプログラムが追加されます。



タスクへのプログラムの割り付け

追加したプログラムの POU を実行させるには、タスクにプログラムを割り付ける必要があります。
追加した Program1 をプライマリ定周期タスクに割り付ける手順は以下のとおりです。

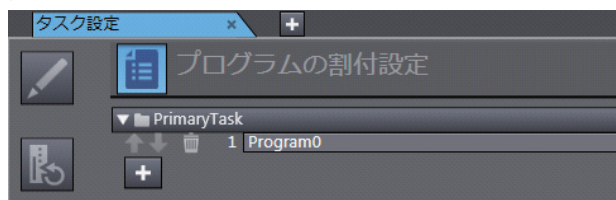
マルチビューエクスプローラで「構成・設定」－「タスク設定」をダブルクリックしてください。



エディットウィンドウの

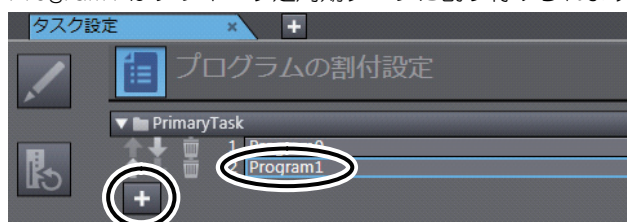
 をクリックしてください。

「プログラムの割付設定」画面が表示されます。



 をクリックして、プルダウンリストから Program1 を選択してください。

Program1 がプライマリ定周期タスクに割り付けられます。

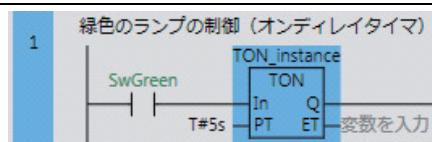


命令の詳細情報の参照

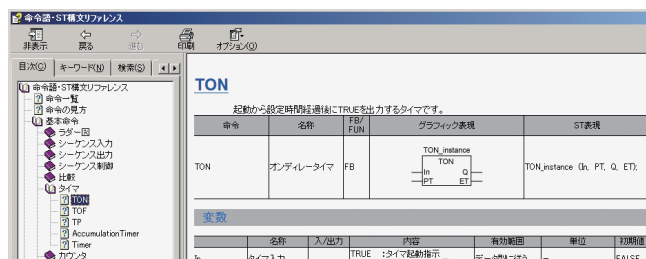
● 命令語リファレンスの表示

命令をクリックして、

を押してください。



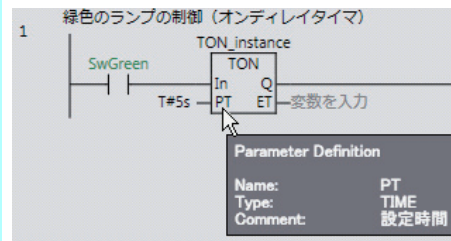
命令語リファレンスが表示されます。



● 命令の入力変数、出力変数の詳細情報の表示

FB 型命令、および FUN 型命令の入力変数、出力変数にマウスカーソルを合わせてください。

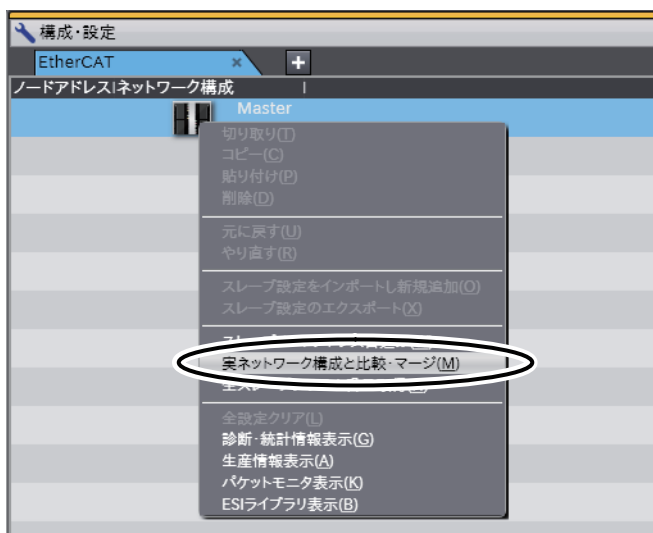
ラダーエディタ上に詳細情報が表示されます。



A-4 オンラインのネットワーク構成の作成

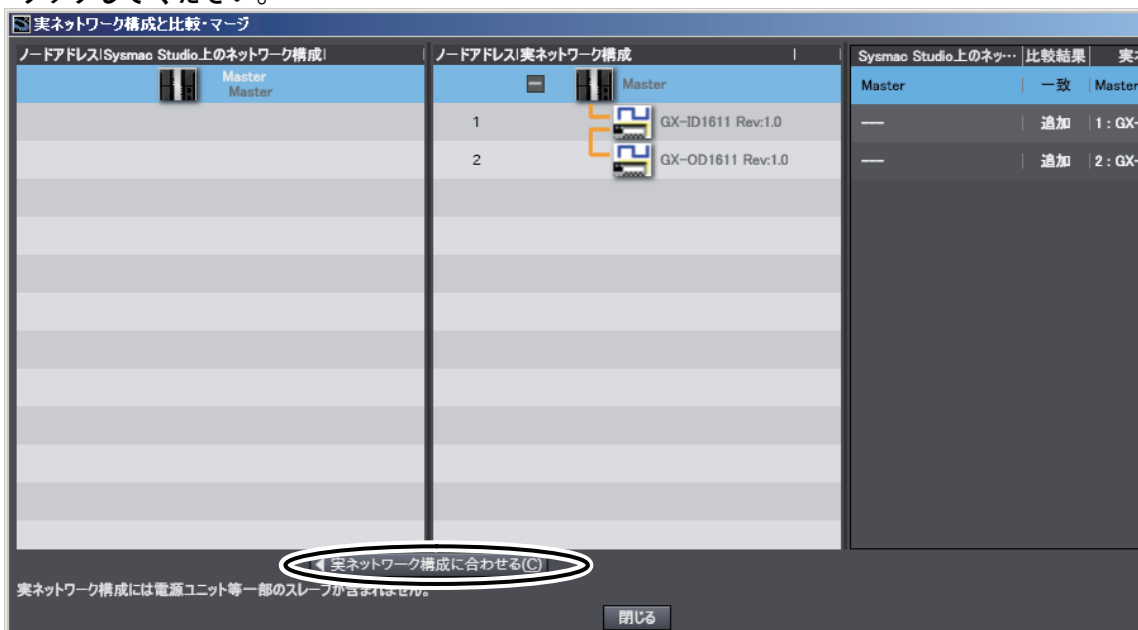
EtherCAT のネットワーク構成の構築が実機で完了している場合は、実機のネットワーク構成から Sysmac Studio 上のネットワーク構成を自動構築することができます。

- 1 オンライン状態で、ネットワーク構成編集ウィンドウのマスタ上で右クリックして「実ネットワーク構成と比較・マージ」を選択してください。

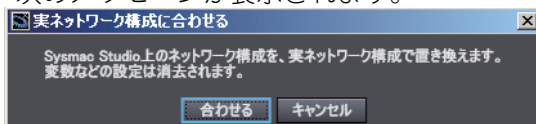


「実ネットワーク構成と比較・マージ画面」が表示されます。

- 2 「実ネットワーク構成と比較・マージ」画面で、「実ネットワーク構成に合わせる」ボタンをクリックしてください。



次のメッセージが表示されます。



3 [合わせる] ボタンをクリックしてください。

実機のネットワーク構成が、Sysmac Studio のネットワーク構成として登録されます。



A-5 オンラインデバッグとオフラインデバッグの違い

オンラインデバッグとオフラインデバッグで、使用可能なデバッグ機能が異なります。デバッグ機能の違いは以下のとおりです。

デバッグ機能	オンラインデバッグ	オフラインデバッグ
モニタリング	○	○
ウォッチウィンドウでのモニタ	○	○
I/O マップでのモニタ	○	○
微分モニタ ^{*1}	○	○
接点のセット/リセット	○	○
接点の強制値リフレッシュ TRUE/FALSE/ 解除	○	○
データの現在値変更	○	○
メモリオールクリア	○	×
クロスリファレンスポップアップ	○	○
オンラインエディット	○	○
コントローラステータスモニタ	○	×
タスク実行状態モニタ	○	○
軸ステータスモニタ (MC モニタテーブル)	○	○
動作モードの変更	○	×
コントローラのリセット	○	×
データトレース	○	○
トリガの設定	○	○
サンプリング変数の設定	○	○
トレースの開始・停止	○	○
トレース結果の表示	○	○
トレース結果のエクスポート	○	○
3D 装置モデルの作成	○	○
デジタル・アナログチャートでの表示	○	○
3D 形式での軌跡表示	○	○
タスク実行時間のモニタ	○	○
実行処理時間予測	×	○
プログラムのシミュレーションデバッグ	×	○
シミュレーションプログラムの設定	×	○
シミュレーションの実行速度の変更	×	○
ブレークポイントの設定	×	○
ステップ実行	×	○
トラブルシュート	○	×
異常情報モニタ	○	○
異常履歴表示	○	○
イベントテーブル設定	○	○
ユーザーメモリの使用量モニタ	○	○
時計情報の設定	○	×
アクセス権の強制開放	○	×

*1. 微分モニタは、CPU ユニット Ver.1.03 以降と Sysmac Studio Ver.1.04 以降の組み合わせで使用可能です。

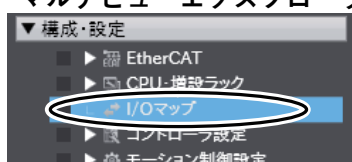
A-6 I/O マップでの強制値リフレッシュ

I/O マップでは、変数の定義やアルゴリズムの作成をしていなくても強制値リフレッシュができます。このため、I/O マップでの強制値リフレッシュは、EtherCAT スレーブや CJ ユニットの入出力の配線を確認するときに便利です。

ここでは、I/O マップでの強制値リフレッシュの操作を説明します。

強制値リフレッシュの操作手順

- 1 オンライン接続してください。または、シミュレータを起動してください。
- 2 マルチビューエクスプローラで [構成・設定] | [I/O マップ] をダブルクリックしてください。



I/O マップが表示されます。

- 3 強制値リフレッシュするI/Oポートまたは変数を、I/O マップ上でクリックして選択してください。

位	ポート	説明	R/W	データ型	値	変数
	▼ CPU・増設ラック					
Ci	CPUラック 0					
	▼ EtherCATネットワークコンフィグレーション					
Et	Master					
Nc	▼ GX-ID1611					
	▼ Read input 1st word	デジタル入力値 (2バイト)		WORD	16#0	
	In Bit00	入力接点00	R	BOOL	FALSE	SwYellow
	In Bit01	入力接点01	R	BOOL	FALSE	SwRed
	In Bit02	入力接点02	R	BOOL	FALSE	SwGreen
	In Bit03	入力接点03	R	BOOL	FALSE	
	In Bit04	入力接点04	R	BOOL	FALSE	
	In Bit05	入力接点05	R	BOOL	FALSE	
	In Bit06	入力接点06	R	BOOL	FALSE	

4 右クリックして [強制値リフレッシュ] – [TRUE] または [FALSE] を選択してください。



[TRUE] を選択すると強制的に ON、[FALSE] を選択すると強制的に OFF されます。選択した I/O ポートまたは変数の [値] 列に、強制状態のマークと、TRUE または FALSE が表示されます。

強制状態を解除する

1 強制状態を解除する I/O ポートまたは変数を選択して右クリックし、[強制値リフレッシュ] – [解除] を選択してください。



I/O ポートまたは変数の強制状態が解除され、強制状態のマークが消えます。I/O ポートまたは変数の ON/OFF の状態は、ユーザプログラムやシステム構成にしたがって変化します。

- 本誌に記載のない条件や環境での使用、および原子力制御・鉄道・航空・車両・燃焼装置・医療機器・娯楽機械・安全機器、その他人命や財産に大きな影響が予測されるなど、特に安全性が要求される用途に使用される際には、当社の意図した商品用途の場合や特別の合意がある場合を除き、当社は当社商品に対して一切保証をいたしません。
- 本製品の内、外国為替及び外国貿易法に定める輸出許可、承認対象貨物(又は技術)に該当するものを輸出(又は非居住者に提供)する場合は同法に基づく輸出許可、承認(又は役務取引許可)が必要です。

オムロン株式会社 インダストリアルオートメーションビジネスカンパニー

●製品に関するお問い合わせ先

お客様相談室

 フリー通話 **0120-919-066** クイック オムロン

携帯電話・PHS・IP電話などではご利用いただけませんので、下記の電話番号へおかけください。

電話 **055-982-5015** (通話料がかかります)

■営業時間：8:00～21:00 ■営業日：365日

●FAXやWebページでもお問い合わせいただけます。

FAX **055-982-5051** / www.fa.omron.co.jp

●その他のお問い合わせ

納期・価格・サンプル・仕様書は貴社のお取引先、または貴社担当オムロン販売員にご相談ください。
オムロン制御機器販売店やオムロン販売拠点は、Webページでご案内しています。

オムロン制御機器の最新情報をご覧ください。

www.fa.omron.co.jp

緊急時のご購入にもご利用ください。

オムロン商品のご用命は